



Banner Workflow

Functional Integration Guide

Release 8.0
December 2008



Banner®, Colleague®, PowerCAMPUS®, Luminis® and Datatel® are trademarks of Ellucian or its affiliates and are registered in the U.S. and other countries. Ellucian, Advance, DegreeWorks, fsaATLAS, Course Signals, SmartCall, Recruiter, MOX, ILP, and WCMS are trademarks of Ellucian or its affiliates. Other names may be trademarks of their respective owners.

©1999-2008 Ellucian. All rights reserved. The unauthorized possession, use, reproduction, distribution, display or disclosure of this material or the information contained herein is prohibited.

Contains confidential and proprietary information of Ellucian and its subsidiaries. Use of these materials is limited to Ellucian licensees, and is subject to the terms and conditions of one or more written license agreements between Ellucian and the licensee in question.

In preparing and providing this publication, Ellucian is not rendering legal, accounting, or other similar professional services. Ellucian makes no claims that an institution's use of this publication or the software for which it is provided will guarantee compliance with applicable federal or state laws, rules, or regulations. Each organization should seek legal, accounting and other similar professional services from competent providers of the organization's own choosing.

Prepared by: Ellucian
4375 Fair Lakes Court
Fairfax, Virginia 22033
United States of America

Revision History

Publication Date	Summary
December 2008	New version that supports Banner Workflow 8.0 software.

Banner Workflow 8.0

Functional Integration Guide

Contents



Section 1 Banner Integration

Chapter 1 Overview

Chapter Summary	9
---------------------------	---

Chapter 2 What Is a Business Event?

How Does a Business Event Start a Workflow?	11
---	----

Who Can Set Up Business Events?	12
---	----

Setting Up Business Events	12
--------------------------------------	----

Workflow Parameter Type Translation	13
---	----

Step 1 - Set Up a Business Event in Banner	14
--	----

Step 2 - Set Up a Business Event in Banner Workflow	15
---	----

Step 3 - Associate the Business Event with Related Workflows in Banner Workflow	16
--	----

Step 4 - Associate a Business Event with a Business Process.	16
--	----

Banner Forms That Support Business Events	17
---	----

Target System Code Validation Form (GTVEQTS)	18
--	----

Event Queue Code Validation Form (GTVEQNM)	19
--	----

Parameter Code Validation Form (GTVEQPM)	20
--	----

Parameter Group Code Validation Form (GTVEQPG)	21
--	----

Parameter Group Rule Form (GOREQPG)	22
---	----

Event Queue Name Definition Form (GOREQNM)	25
--	----

Event Queue Record Maintenance Form (GOAEQRM)27
Key Block27
Events Data Block.28
Parameter Data Block.29
Event Errors Window29

Chapter 3 Processing Banner with Banner Workflow

Processing a Work Item in Your Worklist that Calls a Banner Form31
Communication Between Banner Workflow and Banner32
Characteristics of a Form Called as a Workflow Activity33
Banner Workflow Toolbar Buttons35
Workflow Submit35
Workflow Continue36
Workflow Release36
Other Banner Workflow Processing Considerations37
Navigating From a Form Launched as a Banner Workflow Activity.37
Performing a Rollback37
Exiting a Form Launched as a Banner Workflow Activity38
Exiting or Logging Off Workflow38
Exiting a Banner Session That Was Started by Banner Workflow38
Handling Banner Error Messages While Processing a Banner Workflow Activity39
Using the Process Submission Control Form (GJAPCTL) With Banner Workflow39
Mapping parameters on GJAPCTL39
Performing a Workflow Submit Within GJAPCTL.40

Chapter 4 Example Workflows

Examples.41
New Gifts41
Roles42
Workflow Initiation.42
Workflow Model42
Processing Details44
Database Objects47

Requisition Approvals with E-mail49
Roles49
Workflow Initiation49
Workflow Model49
Processing Details51
Database Objects52
Hire a Full Time Faculty Member54
Roles54
Workflow Initiation55
Workflow Model55
Processing Details57
Database Objects59
Hire a Full Time Staff Member61
Roles61
Workflow Initiation62
Workflow Model62
Processing Details64
Database Objects66
Establish Benefits68
Roles68
Workflow Initiation68
Workflow Model68
Processing Details70
Database Objects72
Load Faculty Information73
Roles73
Workflow Initiation73
Workflow Model73
Processing Details74
Database Objects75
Establish Taxes and Other Deductions76
Roles76
Workflow Initiation76
Workflow Model77
Processing Details78
Database Objects79
New Employee Security Setup81
Roles81
Workflow Initiation81
Workflow Model82
Processing Details83
Database Objects84

Withdraw a Student86
Roles86
Workflow Initiation86
Workflow Model86
Processing Details88
Database Objects89
Process Withdrawn Student Aid90
Roles90
Workflow Initiation90
Workflow Model91
Processing Details92
Database Objects93
Interview Candidate95
Roles95
Workflow Initiation95
Workflow Model96
Processing Details96
Customizing Example Workflows97

Section 2 Luminis Integration

Chapter 5 Luminis

Banner Workflow Product Tab Access101
Add a Banner Workflow Tab102
Banner Workflow Channels102

Section I

Banner Integration





The Banner Workflow Functional Integration Guide is intended for use by the business analyst. It contains information concerning the use of systems that have been integrated with Banner Workflow such as Banner and Luminis.

Chapter Summary

The Banner Integration section contains the following chapters:

- *Chapter 2* - [“What Is a Business Event?”](#) on page 11
- *Chapter 3* - [“Processing Banner with Banner Workflow”](#) on page 31
- *Chapter 4* - [“Example Workflows”](#) on page 41

The Luminis Integration section contains the following chapter:

- *Chapter 5* - [“Luminis”](#) on page 101





A business event is an external stimuli that changes the database, which in turn inserts a record on the Event Queue table. Adding an email address, changing a grade, hiring an employee, and changing a password are all examples of business events.

Banner uses several methods to capture and insert information about a business event on the Event Queue table:

- Database triggers (the most common method)
- Logic located directly in the Banner application
- Any other valid PL/SQL calling block

**Note**

For more information on Business Events, please see the Business Events chapter in the Analyst/Administrator Handbook.

How Does a Business Event Start a Workflow?

Some business events start a workflow automatically. For example, entering new hire information on the Electronic Personnel Action Form (NOAEPAF), under certain circumstances, can automatically start a workflow that grants security to the user.

Here is how a business event is created and how it starts a workflow:

1. Data in the Banner database is added, changed, or deleted.
2. If the database change is set up in Banner to fire a business event, the event code and its associated parameters are captured and stored in the Event Queue table.

In most cases, a database trigger is used to capture and store this information. Another way is to use logic located directly in the Banner application.

3. The Event Dispatcher polls the Event Queue table regularly. If the target system for the business event is WORKFLOW, the Event Dispatcher converts the Banner event into an External Event and posts it to the Banner Workflow system.
4. Banner Workflow evaluates the External Event and its associated parameters to determine which workflows should be started.

Any number of business events can start the same workflow. Similarly, a single business event can start any number of different workflows.



Note

For more information on External Events and how Banner Workflow processes them, please see the External Events chapter in the Technical Integration Guide. ■

Who Can Set Up Business Events?

Only authorized users can set up business events. You should limit these authorized users to system administrators and persons who have complete access to your Banner System and Oracle user maintenance functions. All functionality is delivered with the standard `BAN_DEFAULT_M` access in the General class security. Review this security setup to ensure that only authorized users have access.

Setting Up Business Events

The basic steps involved in setting up a business event in Banner and Banner Workflow is as follows:

1. Set up a business event in Banner.
2. Set up a business event in Banner Workflow.
3. Associate the business event with related workflows in Banner Workflow. This includes mapping the business event parameters to the workflow context parameters.
4. Assign the business event to a business process in Banner Workflow. This includes creating optional guard conditions (rules) that determine when the business event should start each related process.

Workflow Parameter Type Translation

Parameter values, regardless of their Oracle data types, are transferred to the Banner Workflow system as strings. Therefore, care must be taken when transferring values from Oracle to Workflow parameters. Use the following information to determine how to properly construct your SQL select statements.

Type	Details
Text	Since Oracle data, regardless of its type, is transferred to the Banner Workflow system as a string, any Oracle data type that can be returned by a SQL select statement in SQL Plus, can be mapped to a Workflow parameter of type Text.
Boolean	<p>When workflow tries to cast a string to a Boolean value, the String 'true' is converted to true and all other string values are converted to false.</p> <p>Because early versions of Oracle 8i did not have a Boolean data type, Boolean values are commonly represented as a VARCHAR2(1) or CHAR(1) with 'Y' denoting true and 'N' denoting false. To convert this data to a string value of 'true' or 'false' use a SQL decode statement.</p> <pre>decode(<column>, 'Y', 'true', 'N', 'false') or decode(<column>, 'Y', 'true' 'false');</pre>
Numeric	Banner Workflow can cast any string representation of a valid number to the Numeric data type. Therefore, when mapping Oracle Number and Long data types to Workflow Numeric parameters, no conversion is required.
Date	<p>For workflow to successfully cast a string to a Date parameter, the string must be in the following format:</p> <pre><DD>-<MMM>-<YYYY> <HH>:<MM>:<SS></pre> <p>For example:</p> <pre>03-MAR-2003 13:00:00</pre> <p>To convert an Oracle date into a string of this format use the to_char function.</p> <pre>select to_char(<column>, 'DD-MON-YYYY HH:MM:SS')</pre>

Step 1 - Set Up a Business Event in Banner

If you want a business event to start a workflow, you must first set up the business event in Banner. This setup allows you to capture information in the Event Queue table when the database changes.



Note

Banner is delivered with all the setups that support the business events in the example workflows. This seed data has *BASELINE* as the user ID. ■

1. Use the Event Queue Code Validation Form (GTVEQNM) to define the business events that will be captured and sent to Banner Workflow.
2. Use the Parameter Code Validation Form (GTVEQPM) to define the parameters that are associated with business events.
3. Use the Parameter Group Code Validation Form (GTVEQPG) to define the parameter groups that are associated with business events.
4. Use the Parameter Group Rule Form (GOREQPG) to assign defined parameters to groups.
5. Use the Event Queue Name Definition Form (GOREQNM) to define specific business events for Banner Workflow. This definition includes associating a parameter group with each business event.
6. Code and enable database triggers that capture and insert business event information into the Event Queue table.

or

Create logic directly in Banner applications to capture and insert business event information in the Event Queue table.

After business events are set up and Banner starts capturing information on the Event Queue table, the system administrator can use the Event Queue Record Maintenance Form (GOAEQRM) to review event records and change their status.

For more information about the forms used to set up business events, see [“Banner Forms That Support Business Events” on page 17](#).

Step 2 - Set Up a Business Event in Banner Workflow

After you set up a business event in Banner, you must set up the business event in Banner Workflow.



Note

For specific instructions on using Banner Workflow, refer to the *Banner Workflow Analyst/Administrator Handbook* ■

1. Select Business Events from Administration.
2. Select Business Event Definitions.
3. Click **Add Business Event Definition**.
4. Enter information for the business event. Follow these guidelines for Banner events:
 - The **Name** identifies the event code of the business event. This must match the default value of the first parameter associated with the business event in Banner. Banner parameters are set up on the Parameter Group Rule Form (GOREQPG).
 - The **Product Type** must match the default value of the second parameter associated with the business event in Banner. Banner parameters are set up on the Parameter Group Rule Form (GOREQPG).
5. Save the event.
6. In the Parameter section of the page, click **Add Event Parameter**. The **Name**, **Type**, and **Guaranteed** fields are required.

If a special target parameter name is defined for the parameter in Banner on GOREQPG, the parameter **Name** must match the target parameter name entered on GOREQPG. If the parameter does not have a special target parameter name on GOREQPG, the parameter **Name** in Banner Workflow must match the parameter code entered on GOREQPG.

The **Guaranteed** field indicates whether the parameter is guaranteed output from the Banner business event.



Note

You must set up all the Banner parameters (defined on GOREQPG) in Banner Workflow, *except* the first three parameters. ■

Step 3 - Associate the Business Event with Related Workflows in Banner Workflow

You must create associations between the business event and all workflows that the business event can start. This association will contain the information needed to map the parameters in the business event to the context parameters in the associated workflows.

The association between a business event and a workflow is created within the business event.



Note

Refer to the Banner Workflow Analyst/Administrator Handbook for specific instructions on using Banner Workflow. ■

1. Select Business Events from Administration.
2. Select Business Event Definitions.
3. Select a Business Event.
4. In the Associated Workflows section of the page, click **Add Workflow Association**.
5. Select the workflow you would like to associate with the event and click Save.
6. In the parameter mapping section of the Workflow Association page, map all event parameters to their equivalent workflow context parameters.

Step 4 - Associate a Business Event with a Business Process

You must associate a business event with a business process in Banner Workflow.

If the business event should start a workflow only under certain conditions, you must create a guard condition (rule) for the business event. This guard condition is defined at the business process level and determines whether the workflow should start. If the guard condition evaluates to True, the workflow starts. If the guard condition evaluates to False, the workflow does not start.



Note

For specific instructions on using Banner Workflow, refer to the *Banner Workflow Analyst/Administrator Handbook*, ■

1. Select Enterprise Management within Administration.
2. Click a **Business Process** name to assign an event to that business process.
3. In the Associated Events section of the page, click **Add Event Association**.

4. Set the target organization to respond to the event.
5. Select a business event.
6. Click **Save Association**.
7. *(Optional)* If the business event should start a workflow only under certain conditions, click the name of the event you just assigned to add a rule-based guard condition.

If a new business process was created for the event assignment, you will need to assign the workflow to be started to the business process.

1. Select Enterprise Management within Administration.
2. Click the **Business Process** name that you want to assign the workflow to.
3. In the Associated Workflows section of the page, click **Add Workflow Association**.
4. Select the workflow you want to assign to the business process and specify the effective date range.
5. Click **Save Association**.

Banner Forms That Support Business Events

The following Banner forms support business events:

- Target System Code Validation Form (GTVEQTS)
- Event Queue Code Validation Form (GTVEQNM)
- Parameter Code Validation Form (GTVEQPM)
- Parameter Group Code Validation Form (GTVEQPG)
- Parameter Group Rule Form (GOREQPG)
- Event Queue Name Definition Form (GOREQNM)
- Event Queue Record Maintenance Form (GOAEQRM)

These forms appear on the Event Queue Maintenance Menu (*GENEQS) under the System Functions/Administration Menu (*GENSYS) in General.

Target System Code Validation Form (GTVEQTS)

Use this form to define the external systems that require notification when a business event occurs. *WORKFLOW* is one of the target codes delivered with Banner. This code should not be changed or deleted.

When you first access GTVEQTS, the form displays all defined target system codes. You can query any displayed field, including **Activity Date**.

Fields	Descriptions / Buttons
Target Code	<p>Code that uniquely identifies an external system that requires notification when a business event occurs. For Banner Workflow, this code is <i>WORKFLOW</i>. Required.</p> <p>Note: Target codes <i>WORKFLOW</i> and <i>PIPELINE</i> are delivered with the system and should not be changed or deleted. (Changes may impact the processing of business events.)</p>
Description	<p>Freeform description of the target system. Maximum length is 30 characters. This field is accessible only if a Target Code is defined for the record. Required.</p> <p>All user IDs can update this field.</p>
User ID	<p>Code that indicates who inserted the record:</p> <p><i>BASELINE</i> SunGard Higher Education delivered the record as seed data or the client inserted the record while logged into Banner as BASELINE. Only BASELINE users can change or delete the Target Code field.</p> <p><i>LOCAL</i> The client inserted the record with a login ID other than BASELINE. Any user ID can change or delete the Target Code field.</p> <p>Display-only.</p>
Activity Date	<p>Date the record was last updated. Display-only.</p>

Event Queue Code Validation Form (GTVEQNM)

Use this form to define the business events that will be captured and sent to Banner Workflow. Adding an email address, changing a grade, hiring an employee, and changing a password are all examples of business events that can be defined on this form.


When you first access GTVEQNM, the form displays all defined event codes. You can query any displayed field, including **Activity Date**.

Fields	Descriptions / Buttons
Event Code	<p>Code that uniquely identifies a business event. This code is matched to the event name in Banner Workflow during the communication process. Required.</p> <p>You can change or delete the Event Code only if both of the following conditions exist:</p> <ul style="list-style-type: none">• The business event was never run. This means the business event was never inserted into the Event Queue table to be selected by the Event Dispatcher.• The business event is not associated with any target system on the Event Queue Name Definition Form (GOREQNM).
Description	<p>Freeform description of the business event. Maximum length is 30 characters. This field is accessible only if an Event Code is defined for the record. Required.</p> <p>All user IDs can update this field.</p>
User ID	<p>Code that indicates who inserted the record:</p> <p><i>BASELINE</i> SunGard Higher Education delivered the record as seed data or the client inserted the record while logged into Banner as BASELINE. Only BASELINE users can change or delete the Event Code field.</p> <p><i>LOCAL</i> The client inserted the record with a login ID other than BASELINE. Any user ID can change or delete the Event Code field.</p> <p>Display-only.</p>
Activity Date	<p>Date the record was last updated. Display-only.</p>

Parameter Code Validation Form (GTVEQPM)

Use this form to define the parameters that are associated with business events. Employee class, email address, student major, and term code are all examples of parameters that can be defined on this form.

When you first access GTVEQPM, the form displays all defined parameter codes. You can query any displayed field, including **Activity Date**.

Fields	Descriptions / Buttons
Parameter Code	<p>Code that uniquely identifies a parameter. This code is matched to the context parameters in Banner Workflow during the communication process. Required.</p> <p>You can change or delete the Parameter Code only if the parameter code is not associated with any parameter group code on the Parameter Group Rule Form (GOREQPG).</p> <p> Warning Parameter codes are case-sensitive. They must be defined the same way in Banner and in Banner Workflow. ■</p>
Description	<p>Freeform description of the parameter code. Maximum length is 30 characters. This field is accessible only if a Parameter Code is defined for the record. Required.</p> <p>All user IDs can update this field.</p>
User ID	<p>Code that indicates who inserted the record:</p> <p><i>BASELINE</i> SunGard Higher Education delivered the record as seed data or the client inserted the record while logged into Banner as BASELINE. Only BASELINE users can change or delete the Parameter Code field.</p> <p><i>LOCAL</i> The client inserted the record with a login ID other than BASELINE. Any user ID can change or delete the Parameter Code field.</p> <p>Display-only.</p>
Activity Date	<p>Date the record was last updated. Display-only.</p> <p>To display the Activity Date, use the scroll bar at the bottom of the pane that contains the User ID field.</p>

Parameter Group Code Validation Form (GTVEQPG)

Use this form to define the parameter groups associated with business events. For example, a parameter group can be defined for all the parameters passed to Banner Workflow when an employee is hired.

The parameter group code is defined on GTVEQPG. After the code is defined, you can use the Parameter Group Rule Form (GOREQPG) to assign specific parameters to the group.

When you first access GTVEQPG, the form displays all defined group codes. You can query any displayed field, including **Activity Date**.

Fields	Descriptions / Buttons
Group Code	<p>Code that uniquely defines a parameter group. Required.</p> <p>You can change or delete the Group Code only if the group code has no detail records on the Parameter Group Rule Form (GOREQPG).</p>
Description	<p>Freeform description of the group code. Maximum length is 30 characters. This field is accessible only if a Group Code is defined for the record. Required.</p> <p>All user IDs can update this field.</p>
User ID	<p>Code that indicates who inserted the record:</p> <p><i>BASLINE</i> SunGard Higher Education delivered the record as seed data or the client inserted the record while logged into Banner as BASELINE. Only BASELINE users can change or delete the Group Code field.</p> <p><i>LOCAL</i> The client inserted the record with a login ID other than BASELINE. Any user ID can change or delete the Group Code field.</p> <p>Display-only.</p>
Activity Date	<p>Date the record was last updated. Display-only.</p>

Parameter Group Rule Form (GOREQPG)

Use this form to assign defined parameters to groups. For example, you can assign the ID, position code, effective date, and employee class parameters to a group.

After parameters are assigned to a parameter group, you can use the Event Queue Name Definition Form (GOREQNM) to assign the parameter group to business events.

You can insert or delete parameters into a parameter group only if the parameter group is not associated with any target systems on the Event Queue Name Definition Form (GOREQNM).

The first three parameters in a parameter group used by a Banner Workflow business event are required and must be in this order:

1. **EVENTNAME.** The **Default Value** for the first parameter must be the event code of the business event associated with the parameter group. This name must match the event name used in Banner Workflow.
2. **PRODUCTTYPE.** The **Default Value** for the second parameter must match the product type set up for the business event in Banner Workflow. Seed data has *Banner* as the product type, but it can be changed.
3. **WORKFLOWSPECIFICNAME.** This name appears in the worklist when the business event starts a workflow. This name can be defined on this form or in the database trigger coded for the business event.

For example, the parameter group used with the SGHE_B8 New Gift business event might be defined with the following values for the first three parameters:



Parameter Name	Default Value
EVENTNAME	NEWGIFT
PRODUCTTYPE	Banner
WORKFLOWSPECIFICNAME	Donor: John Dickinson made a new gift # 0001543

Fields	Descriptions / Buttons
Group Code	Code that uniquely defines a parameter group. List Parameter Group Validation List (GTVEQPG)
Seq#	Sequence number that defines the order in which the parameter is sent to the target system. Note: The first three parameters in a parameter group used by a Banner Workflow business event are required and must be in a specific order.
Parameter	Code that uniquely identifies a parameter. Required. The first three parameters in a parameter group used by a Banner Workflow business event must be as follows: <ul style="list-style-type: none"> • First parameter — <i>EVENTNAME</i> • Second parameter — <i>PRODUCTYPE</i> • Third parameter — <i>WORKFLOWSPECIFICNAME</i>
	List Parameter Validation List (GTVEQPM)



Warning

Parameter codes are case-sensitive. They must be defined exactly the same in Banner and in Banner Workflow. ■

Fields	Descriptions / Buttons
Default Value	<p>Default value for the parameter. Maximum length is 2000 characters. This default value, if defined, is sent to Banner Workflow. Enter this optional value if the parameter value is the same each time the business event occurs.</p> <p>For the first parameter (EVENTNAME), the default value must be the event code of the business event associated with the parameter group.</p> <p> Warning Default values are case-sensitive. ■</p>
Target Parameter Name	<p>Name used in Banner Workflow to identify the parameter. This field links the Banner parameter with the associated parameter in Banner Workflow. Maximum length is 100 characters.</p> <p>This is an optional field. You can leave this field blank if Banner Workflow uses the name that is in the Parameter field. If Banner Workflow uses another parameter name, you must enter that name in this field. By using this field, you can set up a more user friendly name in Banner Workflow.</p> <p>This field must be blank for the first three parameters.</p> <p> Warning Target parameter names are case-sensitive. They must be defined exactly the same in Banner and in Banner Workflow. ■</p>

Event Queue Name Definition Form (GOREQNM)

Use this form to define specific business events for Banner Workflow. This definition includes associating a parameter group with each business event.

For example, you can use this form to set up a business event for hiring an employee and identify the specific group of parameters associated with that business event.

Fields	Descriptions / Buttons
Target System	Code that uniquely identifies the target system (<i>WORKFLOW</i>). List Target System Validation List (GTVEQTS)
Event Code	Code that uniquely identifies a business event. Required. You can update or delete this code only if the business event was never run. This means the business event was never inserted into the Event Queue table to be selected by the Event Dispatcher. List Event Validation List (GTVEQNM)
Description	Description associated with the event code. Display-only.
Group Code	Code that uniquely identifies a parameter group. Required. You can update or delete this code only if the associated business event was never run. This means the business event was never inserted into the Event Queue table to be selected by the Event Dispatcher. List Parameter Group Validation List (GTVEQPG)
Description	Description associated with the parameter group. Display-only.

Fields	Descriptions / Buttons
Active	<p>Checkbox that determines whether the business event is active or inactive.</p> <p><i>checked</i> The business event is active. Records are inserted into the Event Queue table when the business event occurs. The Event Dispatcher sends parameter values to the appropriate target system for processing.</p> <p><i>unchecked</i> The business event is inactive. Records are not inserted into the Event Queue table when the business event occurs.</p> <p>Business events in seed data are delivered as <i>Inactive</i>. You can activate events when setup is complete.</p>
User ID	<p>Code that indicates who inserted the record. Display only.</p> <p><i>BASELINE</i> SunGard Higher Education delivered the record as seed data or the client inserted the record while logged into Banner as BASELINE. Only BASELINE users can delete these records, change the Event Code field, or change the Group Code field.</p> <p><i>LOCAL</i> The client inserted the record with a login ID other than BASELINE. Any user ID can delete these records, change the Event Code field, or change the Group Code field.</p> <p>To display the User ID, use the scroll bar at the bottom of the pane that contains the Description and Active fields.</p>

Event Queue Record Maintenance Form (GOAEQRM)

Use this form to review event records created in Banner. You can change the status of event records on this form. This form lists all business events by target system. You can further refine the listing by event code and/or event status.

The form shows the following kinds of information:

- **Event Data.** Basic information about all business events that match the criteria in the key block.
- **Parameter Data.** Parameter information for a specific business event selected in the Events Data block.
- **Event Errors.** All error messages for the target system specified in the key block.

Key Block

Use this block to identify the business events you want to review. You must identify the target system you want to review. Within the selected target system, you can optionally refine the list by event code and/or event status.

Fields	Descriptions / Buttons
Target System	Code that uniquely identifies the target system. Required. List Target System Validation List (GTVEQTS)
Event Code	Code that uniquely identifies a business event. List Event Validation List (GTVEQNM) If you enter a value in this field, only records with that event code are displayed in the Events Data block. If you leave the field blank, all business events for the target system are displayed.
Status	Drop-down list that identifies the status of business events. If you select a status, only records with that status are displayed in the Events Data block. If you wish to view all records, select <i>None</i> from the list.

Events Data Block

Use this block to review and/or change the status of business events. This block shows basic information about all business events that match the criteria in the key block.

Sequence #	Unique one-up number generated by the system to identify a business event. Display-only.
Event	Code that uniquely identifies the business event associated with the sequence number. Display-only.
Status	<p>Drop-down list that identifies the current status of the business event. Valid values are as follows:</p> <p><i>Ready for Processing</i> This is a new event that the Event Dispatcher has not selected for processing. The system changes this status to <i>Processing</i> when the Event Dispatcher selects the event. You can also change the status of this event.</p> <p><i>Processing</i> The Event Dispatcher has selected the event but has not yet completed processing it. You can change the status of this event. The system can also change this status to <i>Processed</i> or <i>Rejected</i>.</p> <p><i>Processed</i> The Event Dispatcher has selected the event and has completed processing it. You can change the status of this event. The system does not change this status.</p> <p><i>Rejected</i> The Event Dispatcher has rejected the event and will not process it. You should correct the event or event rules so that the event can be processed. After corrections are made, the system administrator can change the status to the appropriate value. This value is reserved for system use only. The systems administrator cannot set an event to <i>Rejected</i>.</p> <p><i>Hold</i> The event should not be selected by the Event Dispatcher for processing. Only the systems administrator can change this status.</p>
User ID	Oracle ID of the person who last updated the record. Display-only.
Activity Date	Date the record was last updated. Display-only.

Parameter Data Block

Use this block to display parameter information for a specific business event selected in the Events Data block. All fields in this block are display-only.

Fields	Descriptions / Buttons
Sequence #	Number that defines the order in which the parameter is sent to the target system.
Parameter	Code that uniquely identifies a parameter.
Value	Value the system generated for the associated parameter when the business event occurred.

Event Errors Window

Use this window to display all error messages for the target system specified in the key block. All fields in this block are display-only.

Event Code	Code that uniquely identifies a business event.
Error Seq#	Unique one-up number generated by the system to identify the error message.
User	Oracle ID of the person who last updated the record that created the error message.
Activity Date	Date the record was last updated.
Error Message	Oracle error message.
Call Stack	Location in the program code where the error occurred.





This chapter:

- Describes how Banner interacts with Banner Workflow when processing an activity that calls a Banner form.
- Explains the characteristics of a Banner form called as a workflow activity.
- Describes the functions of the Banner Workflow toolbar buttons.
- Explains how to use the Process Submission Control Form (GJAPCTL) with Banner Workflow.

Processing a Work Item in Your Worklist that Calls a Banner Form

1. Navigate to the worklist either by logging into Banner Workflow directly or viewing the worklist as a channel within Luminis.
2. Start the work item by clicking the workflow name. You can start any Available or Reserved work item. If the work item is associated with a Banner form, Banner Workflow checks to see whether an active Banner session was already launched through Banner Workflow and is waiting at the menu form GUAGMNU.
3. INB will be launched.

If Banner Workflow has already launched a Banner session and no forms are currently open in that session, the workflow activity continues. If you are located on the GUAGMNU form, Banner receives the message to perform the workflow activity, obtains the name of the Banner object associated with the workflow activity, validates that it is a valid Banner object and a supported object type, checks security to ensure that you have access, and opens the form or process that corresponds to the workflow activity.



Note

Currently, forms and processes (objects types *FORM* and *JOBS* in the GUBOBS table) are the Banner object types that can be launched as workflow activities. If a workflow activity tries to launch a Banner object with an invalid name or invalid object type, Banner will display an appropriate error message. ■

Communication Between Banner Workflow and Banner

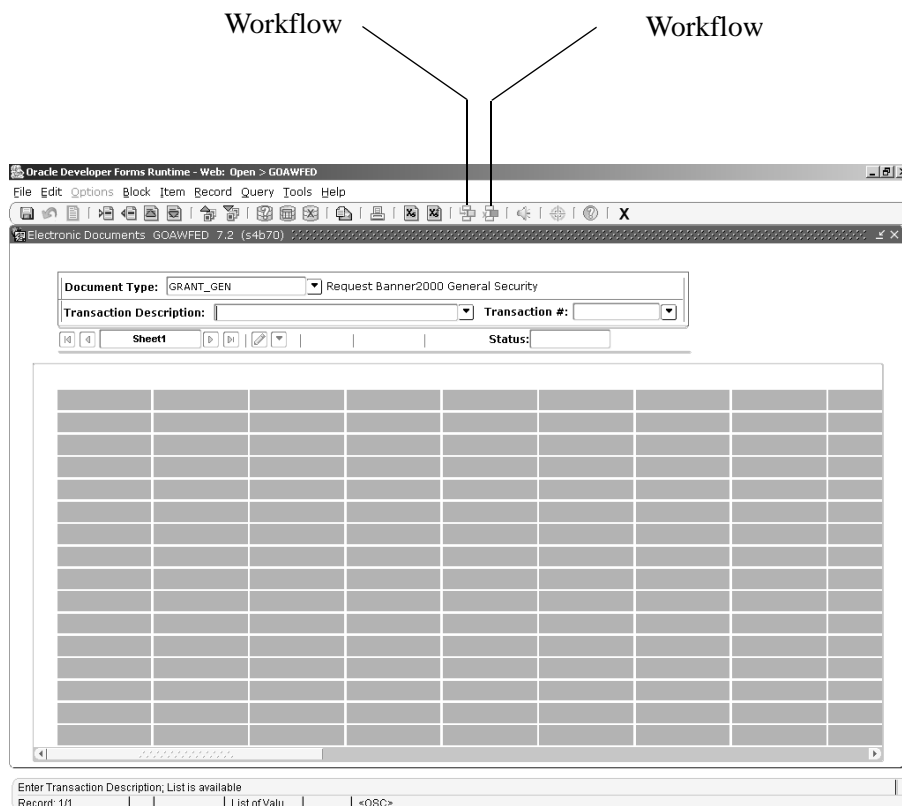
When Banner Workflow opens a Banner session, communication is established between Banner Workflow and Banner. Banner immediately begins to poll for requests to perform Banner Workflow tasks and activities if the appropriate conditions are met.

The following conditions must be met to establish communication between Banner and Banner Workflow:

- Communication between Banner and Banner Workflow must be institutionally enabled (the **Workflow Enabled** indicator on the Installation Control Form (GUAINST) is checked).
- Communication between Banner and the Banner Workflow server must be available.
- The object that serves as the Banner Adapter to Banner Workflow must be located. See Banner Workflow-Awareness Library (GOQWFLW) in Chapter 2 - Banner Integration of the Banner Workflow Technical Integration Guide for more details.

If any of the above conditions are not met, an error message explains why communication could not be established.

Characteristics of a Form Called as a Workflow Activity



When Banner first opens a form as a workflow activity based on a request from Banner Workflow, the following processing occurs:

- The name of the workflow activity associated with the form is attached to the MDI application window title.

The name of the workflow activity appears in brackets after the Banner form title to emphasize that this form is currently part of a workflow activity.

- The **Workflow Submit** and **Workflow Release** buttons are enabled.

This underscores the fact that the current form is a workflow activity, and enables you to perform the required functions for processing the activity. See [“Banner Workflow Toolbar Buttons” on page 35](#) for more details.

- Any parameter names and values that are associated with the workflow activity are retrieved from Banner Workflow and validated.

If any of the parameter names do not match the names of valid items on the form in block.item format, or if one or more form items that are parameters for the current workflow activity can never be populated by their associated input parameter values, the appropriate error message appears.

A form item must meet the following Banner conditions if it is ever to be populated with an input parameter value:

- The item must be associated with a canvas.
- The item must exist in the key block of the form.

In other words, it is safe to assume that an item can automatically be populated by Banner because it meets the basic conditions under which you can populate it upon first entering a form.

When you are navigated to the first block (typically the key block) of a Banner form that has been opened as a workflow activity, Banner determines whether there are any key block items on the form that should be populated with the input parameter values retrieved from Banner Workflow. If there are any such key block items, the following processing occurs:

1. Banner checks each item in the key block of the form and, if that item matches a parameter that has an input parameter value, it:
 - Validates that the item can indeed be populated with a value given its current properties.
 - Reformats the parameter value so it is appropriate for the item.
 - Ensures that the input focus is placed in the item.
 - Populates the item with the appropriate value.
 - Validates the item.

If the navigation to, population of, or validation of any key block item encounters errors, the system will display an appropriate error message.

2. Banner protects from insert or update those key block items that have been populated with input parameter values.
3. Banner highlights key block items that have been populated with input parameter values and protected to distinguish these items from any key block items that have not been populated and protected.

Banner Workflow Toolbar Buttons

Workflow Submit

When you submit a workflow activity, you declare the work complete and indicate that you want Banner Workflow to advance to the next activity in the workflow. Prior to submitting the workflow activity, the status of the activity remains as *Reserved*.

When you click **Workflow Submit**, the following processing occurs:

1. Banner performs a Commit on the form if there are any changes to save and the appropriate conditions are met.

A Commit is performed under the following conditions:

- The form does not follow a naming convention indicating that the form is for inquiry only (that is, the third character of the form name is not *C* or *I*, or the first three letters of the form name are not *FTV*).
- You have not been granted a role (either directly or via a class) for the current form that only allows you to query records (that is, the role associated with your access to the form does not end with *_Q*).

If an error is encountered in Banner's validation of the form to determine whether there are any changes to save, or in its attempt to perform a Commit of the changes made, the appropriate error message will appear.

2. Banner extracts any and all updated parameter values from the form and returns them to Banner Workflow.
3. Banner declares the workflow activity complete to Banner Workflow.
4. Banner unregisters the form as a workflow activity so that it is no longer treated as such by the application.
5. Banner resets all visual cues on the form that indicated that the form was a workflow activity. For example, the Banner Workflow buttons are disabled and the activity name is removed from the Banner MDI application window title.
6. Banner attempts to exit the form and returns to the General Menu Application Form (GUAGMNU).

If the attempt to exit the form fails due to form validation errors, etc., the appropriate error message appears.



Note

A Workflow Submit cannot be performed while a form is in Enter Query mode, as parameter values that have not been validated to exist in the Banner database could erroneously be returned to Banner Workflow. If you click **Workflow Submit** while in Enter Query mode, you will be presented with an error message directing you to either cancel or execute the current query. ■

Once you have submitted a workflow activity from within Banner, the Workflow Status view in Banner Workflow indicates that the activity has been completed and the workflow progresses to the next activity in the model.

Workflow Continue

SunGard Higher Education plans to use the Workflow Continue function in a future release for forms that are part of a QuickFlow that has been launched as a workflow activity. Until then, the **Workflow Continue** button is disabled on all forms under all conditions.

Workflow Release

When you perform a Workflow Release, you are indicating to Banner that you no longer want the current form to be treated as a workflow activity and that you do not want to return the values of any updated parameter values to Banner Workflow at this time so that you can perform other work in Banner.

Specifically, when you click **Workflow Release**, the following processing occurs:

1. Banner unregisters the form as a workflow activity so that it is no longer treated as such by the application.
2. Banner resets all visual cues on the form that indicated that the form was a workflow activity.
 - All Banner Workflow buttons are disabled to prevent the subsequent execution of any Banner Workflow functions.
 - The name of the workflow activity is removed from the Banner MDI application window title.
 - If any key block items were populated with input parameters, protected from insert or update, and highlighted to distinguish the items from other items, the highlight is removed from the items and the items are unprotected so that you can update them as necessary in a non-Banner Workflow context.

Once you have released a workflow activity from within Banner, the Workflow Status view in Banner Workflow indicates that the activity is still in progress and the activity remains in your worklist as *Reserved*.

Other Banner Workflow Processing Considerations

Navigating From a Form Launched as a Banner Workflow Activity

When you navigate from a form that is an active workflow activity to another form, whether it be through Direct Access, Object Search, the navigation frame, or other methods, the Banner Workflow buttons are disabled on the form to which you navigate, and no key block items are protected or highlighted as input parameters on the form. In other words, the form that you open is not a workflow activity.

At the same time, you will have a visual cue that there is an in-progress workflow activity somewhere within Banner because the name of that activity will be visible in the MDI application window title.

When you return to the form that is the active workflow activity, the appropriate Banner Workflow buttons are again enabled, and the appropriate key block items are highlighted and protected.

Performing a Rollback

When you perform a rollback in a form that is a currently active workflow activity, the following processing occurs:

1. Banner temporarily turns off protection on those key block items that have been populated with input parameter values.

The protection of the items from insert or update needs to be temporarily turned off so that no errors are encountered when Banner attempts to repopulate items that cannot be updated.

2. Banner repopulates the appropriate key block items with their associated input parameter values.
3. Banner reprotects the appropriate key block items from insert or update.

Exiting a Form Launched as a Banner Workflow Activity

When you exit a form that has been launched as a workflow activity without performing a Workflow Submit or a Workflow Release, you are performing an “implicit release.” In other words, you are still indicating to Banner that you no longer want the current form to be treated as a workflow activity and that you do not want to return any updated parameter values to Banner Workflow at this time so that you can perform other work in Banner. For this reason, exiting a form without performing a Workflow Release has the same effect as performing a Workflow Release and then exiting.

Once you exit a workflow activity from within Banner, the activity remains in your worklist as *Reserved*.

Exiting or Logging Off Workflow

When you exit or log off Banner Workflow, communication with an associated Banner session is severed. If you are situated on the General Menu Application Form (GUAGMNU) when you exit or log off of Banner Workflow, Banner receives a message from Banner Workflow and severs communication immediately. If you are on another form, you will get the message once you navigate back to GUAGMNU.

If you exit Banner Workflow, communication must be severed because there is no Banner Workflow session with which Banner can communicate.

Exiting a Banner Session That Was Started by Banner Workflow

Banner sends a message to Banner Workflow indicating that an active Banner session is no longer available for communication with Banner Workflow. Banner Workflow will display an appropriate alert requesting whether you want to launch a new Banner session if you start another workflow task or activity that involves a Banner form or process.

Handling Banner Error Messages While Processing a Banner Workflow Activity

If you encounter an error in the processing of a Banner form as a workflow activity, you will receive an error message that indicates:

- the source of the error.
- whether a Workflow Release has automatically been performed as a result of the error.
- whether there is any corrective action you can immediately take to resolve the error.

If additional details are available to assist you in resolving the error and Banner can identify your TEMP directory, it will spool the additional information to a log file in that TEMP directory. A follow-up message will indicate the exact name and location of the log file. You can either open the log file with a text editor and use its contents to resolve the error, or you can forward the contents to the appropriate workflow and/or system administrator so that the appropriate corrective action can be taken.

Using the Process Submission Control Form (GJAPCTL) With Banner Workflow

The Process Submission Control Form (GJAPCTL) can function within the context of Banner Workflow as follows:

- You can use GJAPCTL as a workflow activity within itself, meaning that the workflow definition has an executable ID of *GJAPCTL*.
- You can also use GJAPCTL as an indirect activity within a workflow definition. This is the case when a Banner process is specified as the executable ID associated with a workflow activity.

Mapping parameters on GJAPCTL

In the case where the workflow activity has an executable ID corresponding to a Banner process, the job name should *not* be mapped as an input parameter since Banner by default opens GJAPCTL and populates its key block with the name of the job when a job is encountered. However, you can use the job name as an *output* parameter.

In the case where the workflow activity has an executable ID of *GJAPCTL*, the job name may be used as an input and output parameter. In both cases the **Parameter Set** code field may be used as an input and output parameter.

Additional output parameters have been added to the `FORM_HEADER` block. These items are populated when a job is run.

Job name `FORM_HEADER.WF_JOB_NAME`,
One up number `FORM_HEADER.WF_ONE_UP_NO`,
Log output file name `FORM_HEADER.WF_LOG_FILE_NAME`,
Lis output file name `FORM_HEADER.WF_LIS_FILE_NAME`,
Print code `FORM_HEADER.WF_PRNT_CODE`, and
Display/Submit indicator `FORM_HEADER.WF_DISPLAY_SUBMIT_IND`.

All components corresponding to a Banner job defined within the Business Component Catalog of Banner Workflow contains these additional items as parameters. The component representing GJAPCTL also contains these items as parameters.

Performing a Workflow Submit Within GJAPCTL

When GJAPCTL is launched, either directly or indirectly, to perform an activity that you have started from Banner Workflow, you can enter the appropriate job parameters and other information as you typically would before running the job. You can then either select the Workflow Submit or Save function while positioned in the Submission block of the form to submit the workflow activity.

Selecting either Workflow Submit or Save when the cursor is positioned in the Submission block of the form does not merely run the job and return to the form as when the form is not being executed as a workflow activity. Logic has been added such that either selection runs the job, returns to the form, displays a message with details about the submitted job, and finally submits the activity to Banner Workflow, and then exits the form. This logic only occurs when GJAPCTL is used to perform a workflow activity.



SunGard Higher Education delivers Banner example workflows and the seed data that supports them. The purpose of these examples is to show you how workflows can be built and to provide samples of the functionality you may want to use for your own workflows.

Examples

This chapter describes the following example workflows delivered by SunGard Higher Education:



Note

In Banner Workflow 8.0, all example workflows are delivered with a “SGHE_B8” prefix to differentiate the example workflows from other workflows in your system.

- New Gifts
- Requisition Approval with E-mail Functionality
- Hire a Full Time Faculty Member
- Hire a Full Time Staff Member
- Establish Benefits
- Load Faculty Information
- Establish Taxes and Other Deductions
- New Employee Security Setup
- Withdraw a Student
- Process Withdrawn Student Aid
- Interview Candidate

Instructions for customizing these examples are provided at the end of this chapter.

New Gifts

This workflow is initiated whenever a gift is entered into Banner that is greater than or equal to \$500. An automated activity retrieves and passes gift information to the workflow.

Five parallel activities notify the appropriate Alumni/Development administration personnel that a donor has made a new gift:

- The Annual Giving Director is notified if the gift is part of an annual fund.
- The Director of Development is notified if the gift is greater than or equal to \$1000.
- Stewardship is notified if the gift is greater than or equal to \$500.
- If the donor has an assigned Development Officer, that person is notified of the gift so contact information can be updated on the Prospect Comment Form (AMACONT).
- If the gift was a planned gift, the Planned Gift Officer is notified so the planned giving information on the Gift Auxiliary Amount Form (AGAGAUX) can be viewed or updated, and associated ID information on the Gift Associated ID Form (AGAGASC) can be viewed or updated.

Roles

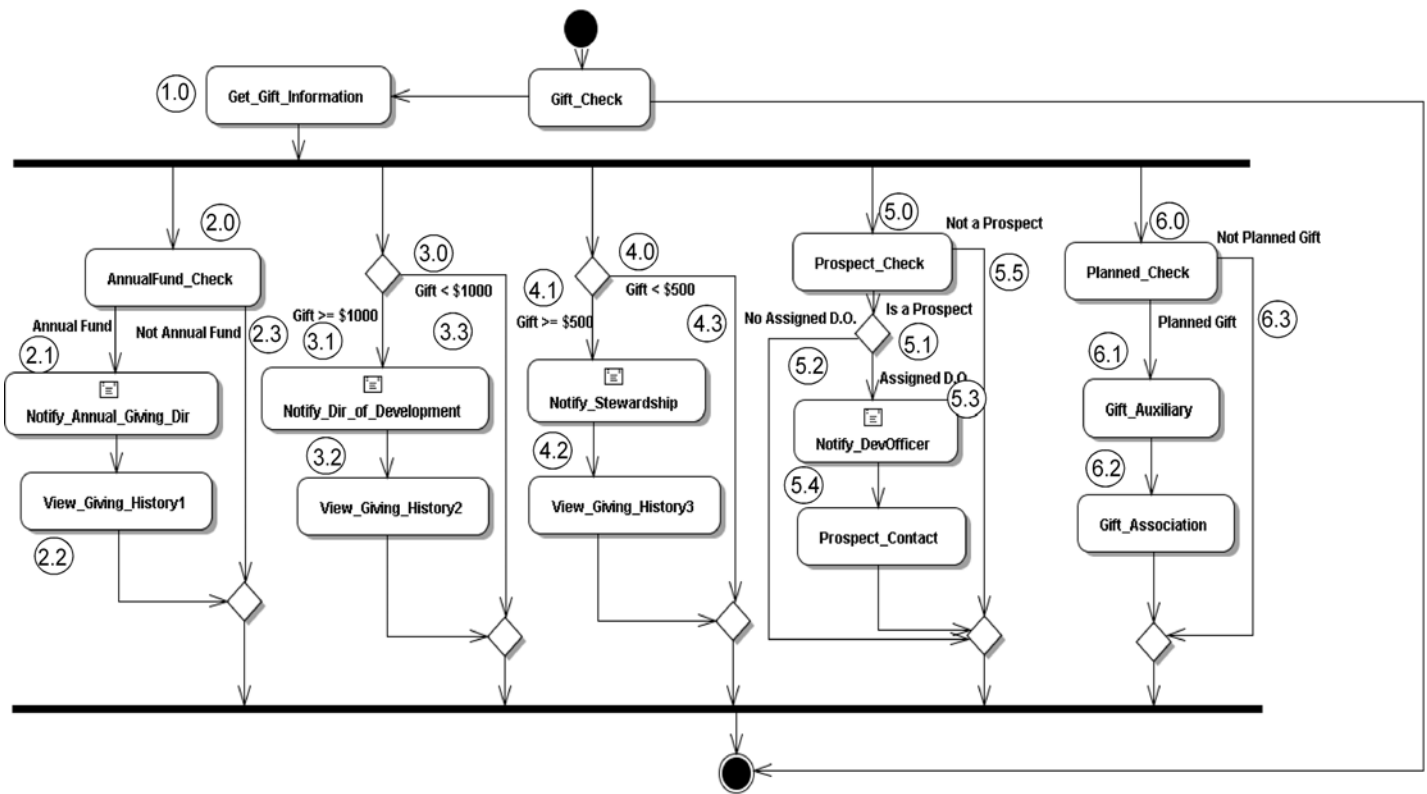
Role	User ID
Stewardship	adisusr
Director of Development	adisusr
Annual Giving Director	adisusr
Development Officer	adisusr
Planned Gift Officer	adisusr

Workflow Initiation

The business event `NEWGIFT` is created from a database trigger on the `AGBGIFT` table. If a new gift amount is greater than or equal to \$500, the business event starts the New Gift workflow. The business event has a guard condition that prevents the workflow from starting if the gift amount is less than \$500.

Workflow Model

The following model illustrates the New Gifts workflow. The circled numbers refer to the processing details that follow.



Processing Details

GET_GIFT_INFORMATION is an automated activity that retrieves stored gift information (1.0). Banner Workflow notifies five roles, each of which has a distinct path in the workflow.

Annual Giving Director Path

An automated activity, ANNUALFUND_CHECK, determines whether the gift/payment is designated for an annual fund campaign, and whether the amount is greater than zero (2.0).



Note

An annual fund campaign has a campaign type code of *AF*. ■

If those two conditions are not met, the path ends, and nothing appears on the Annual Giving Director's worklist (2.3).

If both conditions are met, Banner Workflow sends an e-mail, NOTIFY_ANNUAL_GIVING_DIR, of the gift/payment and donor to the Annual Giving Director (2.1). The e-mail includes the following information:

- Donor ID
- Donor name
- Whether this is the first gift/payment from this donor
- Primary donor category
- Donor email address
- Gift/payment amount
- Gift number
- Gift date
- Onetime gift/pledge payment indicator
- Pledge balance
- Solicitation type

The Annual Giving Director can start the interactive activity, VIEW_GIVING_HISTORY1, to display the Giving History Summary Form (APAGHIS) (2.2). This activity is optional, however, and the **Skip** button can be clicked to continue.

Director of Development Path

A decision determines whether the gift/payment amount (AGBGIFT_AMT_TOT) is greater than or equal to \$1000 (3.0). If it is less than \$1000, the path ends (3.3). If it is greater than or equal to \$1000, Banner Workflow sends an e-mail, NOTIFY_DIR_OF_DEVELOPMENT, to the Director of Development (3.1). This notification includes the following information:

- Donor ID
- Donor name
- Whether this is the first gift/payment from this donor
- Primary donor category
- Donor email address
- Gift/payment amount
- Gift number
- Gift date
- Onetime gift/pledge payment indicator
- Pledge balance
- Solicitation type

The Director of Development can start the interactive activity, VIEW_GIVING_HISTORY2, to display the Giving History Summary Form (APAGHIS) (3.2). This activity is optional, however, and the **Skip** button can be clicked to continue.

Stewardship Path

A decision determines whether the gift/payment amount (AGBGIFT_AMT_TOT) is greater than or equal to \$500 (4.0). If it is less than \$500, the path ends (4.3). If it is greater than or equal to \$500, Banner Workflow sends an e-mail, NOTIFY_STEWARDSHIP, to Stewardship (4.1). This notification includes the following information:

- Donor ID
- Donor name
- Whether this is the first gift/payment from this donor
- Primary donor category
- Donor email address
- Gift/payment amount
- Gift number
- Gift date

- Onetime gift/pledge payment indicator
- Pledge balance
- Solicitation type

Stewardship can start the interactive activity, `VIEW_GIVING_HISTORY3`, to display the Giving History Summary Form (APAGHIS) (4.2). This activity is optional, however, and the **Skip** button can be clicked to continue.

Development Officer Path

An automated activity, `PROSPECT_CHECK`, determines whether the donor is a prospect by checking to see if there is an `AMRINFO` record for that donor (5.0). If not, the path ends (5.5).

A decision determines whether a Development Officer has been assigned to the donor by checking the primary indicator (`AMRSTAF_PRIMARY_IND`) in the database (5.1).

If the primary indicator is not selected, the path ends (5.2).

If a Development Officer is assigned (with a primary indicator) to the donor, Banner Workflow sends an e-mail, `NOTIFY_DEVOFFICER`, to anyone with the role of Development Officer (5.3).

The notification includes the following information:

- Donor ID
- Donor name
- Whether this is the first gift/payment from this donor
- Primary donor category
- Donor email address
- Gift/payment amount
- Gift number
- Gift date
- Onetime gift/pledge payment indicator
- Pledge balance

An interactive activity, `PROSPECT_CONTACT`, displays the Prospect Contact Form (AMACONT) (5.4). This activity is optional, however, and the **Skip** button can be clicked to continue.

Planned Gift Officer Path

An automated activity, `PLANNED_CHECK`, determines whether the gift/payment is a planned gift by checking the gift type code (`AGBGIFT_GIFT_CODE`) (6.0). If the gift type is mapped to a VSE type of *D*, *O*, *R*, or *T*, the gift is a planned gift, and the path continues. Any other gift type causes the path to end (6.3).

If the path continues, an interactive activity, `GIFT_AUXILIARY`, displays the Gift Auxiliary Amount Form (`AGAGAUX`) (6.1) and an interactive activity, `GIFT_ASSOCIATION`, displays the Gift Associated ID Form (`AGAGASC`) (6.2) for the Planned Gift Officer. The Planned Gift Officer can click the **Skip** button to continue.

Database Objects

The New Gifts workflow uses the following database objects:

Triggers				
Name	Description/Where Used	New	Creation script	Dependencies
A_WFAIR_NEWGIFT_INSERT	Conditionally invokes a procedure to create the event NEWGIFT.	Yes	AGTGIFT0.SQL	General 4.1

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
AOKWEVN.P_AGBGIFTNEWGIFT	Creates event NEWGIFT, which conditionally starts the workflow.	Yes	AOKWEV1.SQL AOKWEVN.SQL	General 4.1
AOKWEVN.P_GET_GIFT_INFO	Used in an automated activity to return supplemental information about the gift to display in the manual activities of the workflow.	Yes	AOKWEV1.SQL AOKWEVN.SQL	General 4.1

Functions				
Name	Description/Where Used	New	Creation script	Dependencies
AOKWEVN.F_GIFTEXISTS	Used in a workflow to verify that the new gift still exists.	Yes	AOKWEV1.SQL AOKWEVN.SQL	General 4.1
AOKWEVN.F_AF_FUND	Used in a workflow to determine whether the gift was for an annual fund campaign.	Yes	AOKWEV1.SQL AOKWEVN.SQL	General 4.1
AOKWEVN.F_PLANNED_GIFT	Used in a workflow to determine whether the gift was a planned gift.	Yes	AOKWEV1.SQL AOKWEVN.SQL	General 4.1
F_PROSPECT_IND	A standalone function used in a workflow to determine whether the donor of the gift is a prospect.	No	APFPIND.SQL	General 4.1 and Alumni 2.0.6
APKDPRI.F_SELECT_DONOR_CATG	Used in procedure P_AGBGIFTNEWGIFT to put the donor's primary donor category into the event parameter.	No	APKDPRI.SQL	General 4.1 and Alumni 2.1.11

Requisition Approvals with E-mail

The Requisition Approvals with E-mail example is assigned to the DOCAPPROVE business event and the Generate & Approve Departmental Requests business process.

This workflow sends a requisition through the steps necessary for approval. The workflow determines who the next approver is, and directs an activity to that person's worklist. If that person approves the document, the workflow determines who the next approver is, and the cycle continues until all approvers have reviewed and approved the requisition.

If the document is not approved by any one of the approvers, the Requisition Clerk receives an activity to either correct or delete the requisition. In either case, the workflow is considered finished. The corrected document will trigger a new workflow if it is inserted into the FOBAINP table.

When all approvers have approved the requisition, the requestor receives an e-mail indicating that the requisition has received final approval.

Note

This workflow uses the baseline Banner Finance System Approvals module. You must have the appropriate approvals structure in place and fully operational for Requisition Documents. For information on setting up the Approvals module, refer to chapter 3, *Processing*, in the *Banner Finance System User Manual*. ■

Roles

Role	User ID
Approver	fimsusr
Requisition Clerk	fimsusr

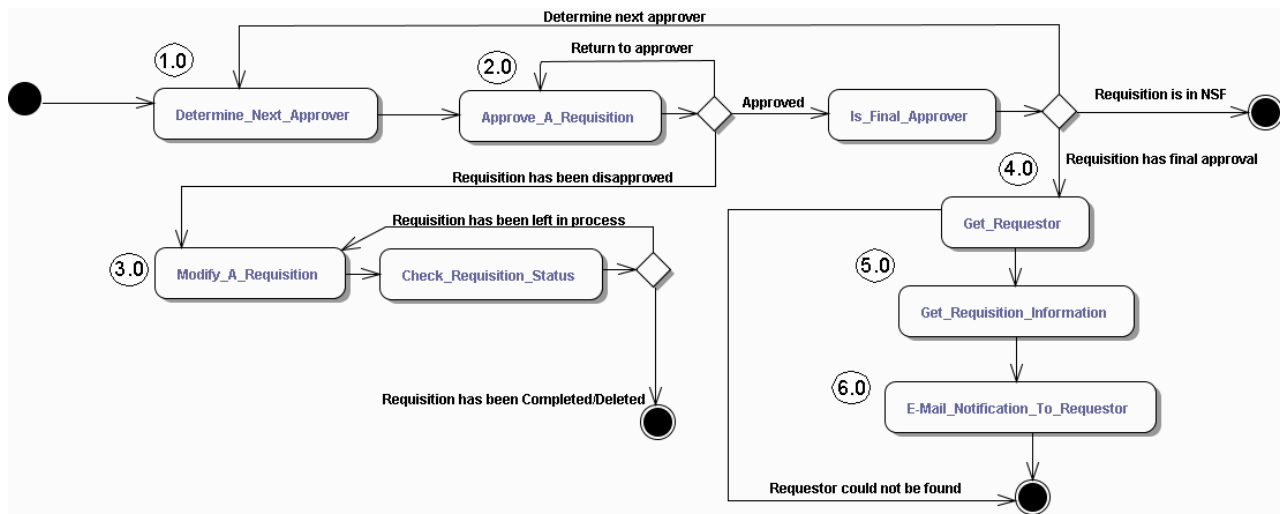
Workflow Initiation

The workflow starts via a business event, DOCAPPROVE, which is triggered when both of the following conditions occur:

- A record is inserted into the FOBAINP table.
- The record has a document number 1 (requisition).

Workflow Model

The following model illustrates the workflow for Requisition Approval with E-mail Functionality. The circled numbers refer to the processing details that follow the model.



Processing Details

An automated activity, `DETERMINE_NEXT_APPROVER`, determines the next approver in the queue for a given document (1.0).

A directed activity, `APPROVE_A_REQUISITION`, is sent to that approver. The approver displays the Document Approval Form (FOAAINP), and approves or disapproves the document, as required (2.0).

If the approver disapproves the document, the Requisition Clerk performs an interactive activity, `MODIFY_A_REQUISITION`, that displays the Requisition Form (FPAREQN) (3.0). The clerk can correct the requisition or delete it. Either way, the workflow ends. If the clerk corrects the document, a new workflow will be created for the corrected requisition when the record(s) are inserted in the FOBAINP table for the NSF queue and FORAPPL is run.

If the approver approves the document, and there are more approvers for the requisition, the cycle begins again (1.0).

If this is the last approver and the requisition is in an NSF (non-sufficient funds) condition, the workflow ends. A new workflow will be created for the same requisition when record(s) are inserted into the FOBAINP table for the NSF queue and FORAPPL is run.

If this is the last approver and the requisition is not in an NSF condition, an automated activity, `GET_REQUESTOR`, returns the originator of the requisition (4.0):

- If a requestor cannot be found, the activity returns the value *Unknown*, and the workflow ends.
- If the requestor is found, an automated activity, `GET_REQUISITION_INFORMATION`, retrieves requisition information (5.0). Then an e-mail activity, `E-MAIL_NOTIFICATION_TO_REQUESTOR`, is sent to the requestor saying that the requisition was approved (6.0). The workflow ends.



Note

As delivered, this example uses FIMSUSR as the User ID for the Requestor for the E-Mail Notification. For FIMSUSR to receive an e-mail message, an e-mail address must be associated with the User ID. This association can be created via User Account Management. ■

Database Objects

The Requisition Approval with E-mail Functionality workflow uses the following database objects:

Triggers				
Name	Description/Where Used	New	Creation script	Dependencies
FT_WFAIR_DOCAPPROVE_INSERT	Fires when a requisition record is inserted into FOBAINP. It is used to fire the business event DOCAPPROVE.	Yes	FOTAINP.SQL	

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
DETERMINE_NEXT_APPROVER	Passes document number and document type into the workflow, and retrieves the next approver for the given document. It is used in an automated activity, DETERMINE_NEXT_APPROVER.	Yes	FOKWFLW.SQL FOKWFL1.SQL	
GET_REQUESTOR	Passes document number into the workflow and retrieves the requestor for a given requisition. It is used in an automated activity called GET_REQUESTOR.	Yes	FOKWFLW.SQL FOKWFL1.SQL	
P_GET_REQ_INFORMATION	Passes document number into the workflow and retrieves requisition information. It is used in an automated activity called GET_REQUISITION_INFORMATION.	Yes	FOKWFLW.SQL FOKWFL1.SQL	

Functions				
Name	Description/Where Used	New	Creation script	Dependencies
IS_FINAL_APPROVER	Passes document number and document type into the workflow. Determines whether the document has any additional approvers that must review it. Used on a transition on an approved requisition.	Yes	FOKWFLW.SQL FOKWFL1.SQL	
CHECK_REQ_STATUS	Passes document number into the workflow. Determines the status of the requisition (<i>Deleted</i> , <i>Complete</i> , or <i>In Process</i>). Used on a transition after the Requisition Clerk has modified/deleted the requisition.	Yes	FOKWFLW.SQL FOKWFL1.SQL	

Hire a Full Time Faculty Member

A member of the Dean's office can use this workflow to create a faculty appointment. The workflow checks the new hire's citizenship and visa, allows the Budget Office to verify that there is enough money in the budget to support the position, and waits for the PAF to be approved by all approvers.

When the PAF is approved and the data is applied to the Banner database, the workflow notifies the following roles simultaneously:

- Employment Manager
- Academic Dean (triggers the Load Faculty Information workflow)
- Benefits Administrator (triggers the Establish Benefits workflow)
- Payroll Manager (triggers the Establish Taxes and Other Deductions workflow)
- Orientation Coordinator
- Registrar (allows assignment of the faculty member in the Banner Student System)

The workflow also triggers the New Employee Security Setup, Load Faculty Information, Establish Taxes and Other Deductions, and Establish Benefits workflows.

Roles

Role	User ID
Employment Manager	hrisusr
Budget Director	fimsusr
	hrisusr
Academic Dean	saisusr
Benefits Administrator	hrisusr
Payroll Manager	fimsusr
	hrisusr
Registrar	saisusr
Orientation Coordinator	hrisusr
Hire Originator	hrisusr

Workflow Initiation

The Dean's office creates a faculty appointment by executing the Electronic Personnel Action Form (NOAEPAF). This insert into the NOBTRAN table creates the PAFCHANGE business event. If the status is *Pending* and the approval category is *FFHIRE*, the workflow starts.

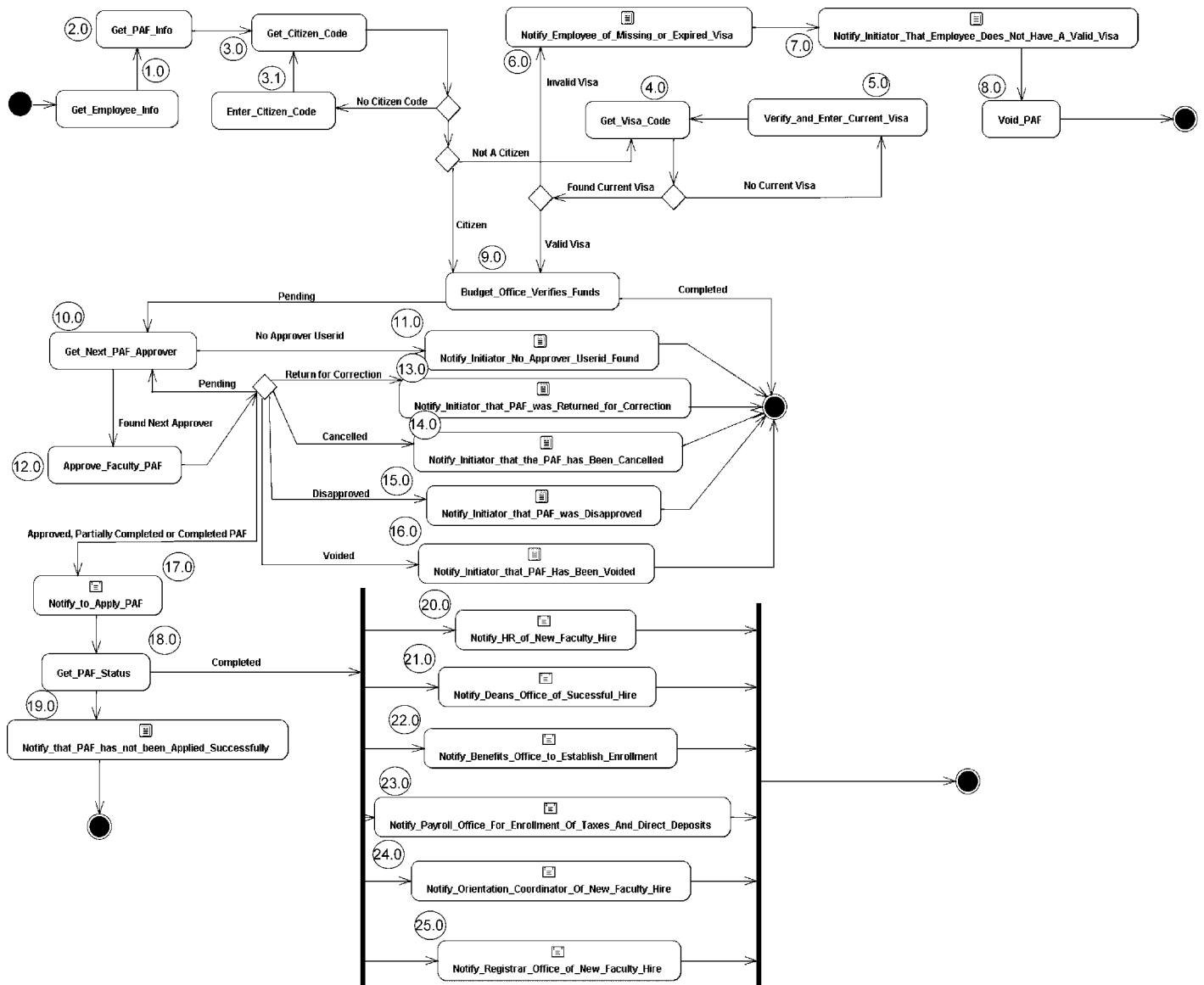


Note

For this example workflow to operate, NOAEPAF must be configured to create the Employee and Job Assignment records. It must also contain *FFHIRE* as the approval category code. ■

Workflow Model

The following model illustrates the Hire a Full Time Faculty Member workflow. The circled numbers refer to the processing details that follow.



Processing Details

The automated activity, `GET_EMPLOYEE_INFO`, retrieves employee information (1.0). The automated activity, `GET_PAF_INFO`, retrieves PAF information (2.0).

The automated activity, `GET_CITIZEN_CODE`, retrieves the citizen code (3.0). If there is no citizen code, an interactive activity displays the Identification Form (PPAIDEN) so the Employment Manager can enter the citizenship information (3.1). After the information has been entered and saved, the citizen code is retrieved, and the workflow continues.

If the new hire is not a U.S. citizen (`SPBPERS_CITZ_CODE = "N"`), an automated activity, `GET_VISA_CODE`, retrieves the visa code (4.0). A business rule checks to see whether the current visa code is valid for working in the U.S. (`SPRINTL_VTYP_CODE_CURRENT = "F1", "B1", "J1", "PR", or "RA"`).

- If the value for the visa code is null, an interactive activity, `VERIFY_AND_ENTER_CURRENT_VISA`, displays the International Information Form (GOAINTL) so the Employment Manager can add it (5.0). Then the workflow retrieves the current visa code (4.0), and the workflow continues.
- If the visa code is invalid, a manual notification, `NOTIFY_EMPLOYEE_OF_MISSING_OR_EXPIRED_VISA`, tells the Hiring Manager to notify the new hire that he/she cannot work in the U.S. (6.0). A manual activity, `NOTIFY_INITIATOR_THAT_EMPLOYEE_DOES_NOT_HAVE_A_VALID_VISA`, tells the Hiring Manager that the hiring process is incomplete and the employment PAF must be voided and resubmitted (7.0). The interactive activity, `VOID_PAF`, displays the Electronic Personnel Action Form (NOAEPAF) so the Hiring Manager can void the PAF (8.0). The workflow ends.
- If the visa code is valid, the workflow continues.

If the new hire is a U.S. citizen (`SPBPERS_CITZ_CODE = "Y"`), the Budget Director uses the Position Budget Form (NBAPBUD) to verify that there is money in the budget to fund the position (9.0).

An automated activity, `GET_NEXT_PAF_APPROVER`, looks for an approver (10.0).

- If no approver is found, a manual activity, `NOTIFY_INITIATOR_NO_APPROVER_USERID_FOUND`, notifies the Hiring Manager that no approver was found and that the new hire needs to be applied manually (11.0). The workflow ends.
- If an approver is found, the interactive activity, `APPROVE_FACULTY_PAF`, sends the NOAEPAF Form to the designated approver (12.0).

The designated approver enters information on the NOAEPAF Form. The next activity depends on the resulting status of the NOAEPAF Form:

- If the PAF is pending, an automated activity, `GET_NEXT_PAF_APPROVER`, looks for the next approver (10.0), and the workflow continues.

- If the PAF is returned for correction, a directed activity, NOTIFY_INITIATOR_THAT_PAF_WAS_RETURNED_FOR_CORRECTION, notifies the Hiring Manager (13.0), and the workflow ends.
- If the PAF is canceled, a directed activity, NOTIFY_INITIATOR_THAT_THE_PAF_HAS_BEEN_CANCELLED, notifies the Hiring Manager (14.0), and the workflow ends.
- If the PAF is disapproved, a directed activity, NOTIFY_INITIATOR_THAT_PAF_WAS_DISAPPROVED, notifies the Hiring Manager (15.0), and the workflow ends.
- If the PAF is voided, a directed activity, NOTIFY_INITIATOR_THAT_PAF_HAS_BEEN_VOIDED, notifies the Hiring Manager (16.0), and the workflow ends.
- If the PAF is approved, partially completed, or completed, the workflow continues.

The e-mail activity, NOTIFY_TO_APPLY_PAF, sends an e-mail to the Hiring Manager that the PAF is ready to be applied (17.0). The Hiring Manager notifies the individual responsible for applying Faculty PAF's to apply the transaction in the Banner database.

An automated activity, GET_PAF_STATUS, retrieves the PAF status (18.0):

- If the PAF is not complete, the directed activity, NOTIFY_THAT_PAF_HAS_NOT_BEEN_APPLIED_SUCCESSFULLY, sends a manual notification to the Hiring Manager that the PAF has not been applied successfully, and the workflow ends (19.0). The originator must create a new transaction from the beginning.
- If the PAF is complete, the following activities are triggered simultaneously:
 - An e-mail activity, NOTIFY_HR_OF_NEW_FACULTY_HIRE, notifies the Employment Manager of the new faculty member (20.0).
 - An e-mail activity, NOTIFY_DEANS_OFFICE_OF_SUCCESSFUL_HIRE, notifies the Academic Dean of the new faculty member (21.0).
 - An e-mail activity, NOTIFY_BENEFITS_OFFICE_TO_ESTABLISH_ENROLLMENT, notifies the Benefits Administrator of the new faculty member (22.0).
 - An e-mail activity, NOTIFY_PAYROLL_OFFICE_FOR_ENROLLMENT_OF_TAXES_AND_DIRECT_DEPOSITS, notifies the Payroll Manager of the new faculty member (23.0).
 - An e-mail activity, NOTIFY_ORIENTATION_COORDINATOR_OF_NEW_FACULTY_HIRE, notifies the Orientation Coordinator of the new faculty member (24.0).
 - An e-mail activity, NOTIFY_REGISTRARS_OFFICE_OF_NEW_FACULTY_HIRE, notifies the Registrar of the new faculty member (25.0). The Registrar can assign the new hire to the Banner Student System outside the workflow.

Database Objects

The Hire a Full Time Faculty Member workflow uses the following database objects:

Triggers				
Name	Description/Where Used	New	Creation script	Dependencies
NT_WFAUR_PAFCHANGE_UPDATE	Inserts a row into event queue when status on PAF status table NOBTRAN changes to <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN0.SQL	
NT_WFAIR_PAFCHANGE_INSERT	Inserts a row into event queue when a new row is inserted on PAF status table and its status is <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN1.SQL	

Functions				
Name	Description/Where Used	New	Creation script	Dependencies
F_GET_NOBTRAN_STATUS	Determines the PAF's status after the notification to apply the PAF. Depending on the PAF's status, the workflow branches to completion or continues.	Yes	NOKMISC.SQL	

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
POKWLIB.P_GET_EMPL_INFO	Gets the person's name, employee class, and home department.	Yes	POKWLIB.SQL POKWLI1.SQL	Payroll must be present.
POKWLIB.P_GET_PAF_POSN_INFO	Gets the PAF position description, originator userid, and originator name.	Yes	POKWLIB.SQL POKWLI1.SQL	Payroll must be present.

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
P_NOBTRANPAFCHANGE	Package called by the above trigger to insert the appropriate data into the Event Queue tables.	Yes	NOKWEVN.SQL NOKWEV1.SQL	
P_GET_NEXT_PAF_USERID	Gets the next userid that needs to approve the PAF and information on whether this ID was obtained. Fired by the automated activity if the ID is eligible for hiring.	Yes	NOKWLIB.SQL NOKWLI1.SLQ	
P_GET_CITIZEN_CODE	Gets the citizen code.	Yes	POKWLIB.SQL POKWLI1.SQL	
P_GET_VISA_CODE	Gets the visa code.	Yes	POKWLIB.SQL POKWLI1.SQL	

Hire a Full Time Staff Member

A member of the Hiring office can use this workflow to create a staff appointment. The workflow checks the new hire's citizenship and visa, allows the Budget Office to verify that there is enough money in the budget to support the position, and waits for the PAF to be approved by all approvers.

When the PAF is approved and the data is applied to the Banner database, the workflow notifies the following roles simultaneously:

- Employment Manager
- Benefits Administrator(triggers the Establish Benefits workflow)
- Payroll Manager (triggers the Establish Taxes and Other Deductions workflow)
- Orientation Coordinator

The workflow also triggers the New Employee Security Setup, Establish Benefits, and Establish Taxes and Other Deductions workflows.

Roles

Role	User ID
Employment Manager	hrisusr
Budget Director	fimsusr
	hrisusr
Human Resources Director	hrisusr
Benefits Administrator	hrisusr
Payroll Manager	fimsusr
	hrisusr
Orientation Coordinator	hrisusr
Hire Originator	hrisusr

Workflow Initiation

The Hiring office creates a staff appointment by executing the Electronic Personnel Action Form (NOAEPAF). This insert into the NOBTRAN table creates the PAFCHANGE business event. If the status is *Pending* and the approval category is *FSHIRE*, the workflow starts.

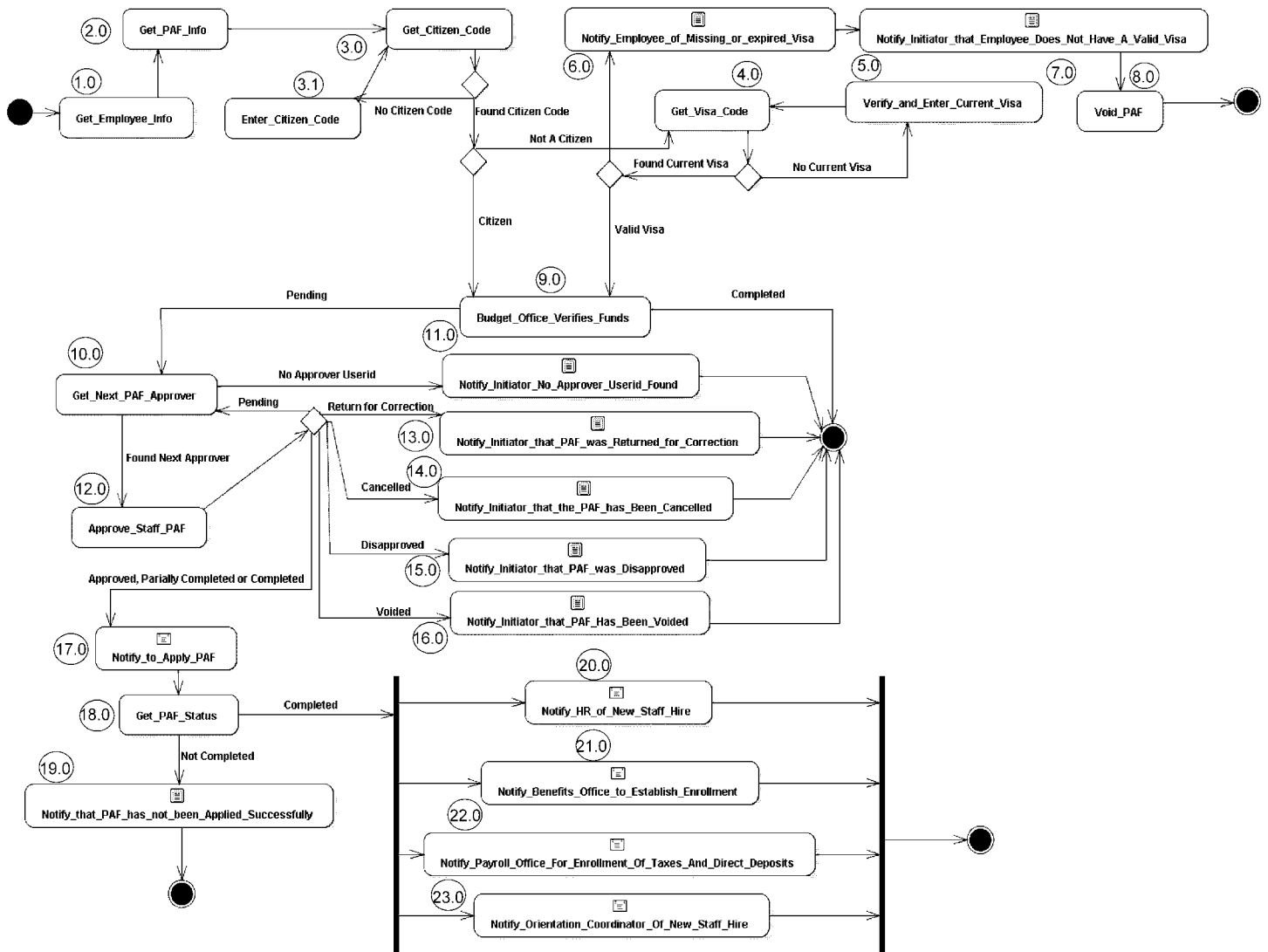


Note

For this example workflow to operate, NOAEPAF must be configured to create the Employee and Job Assignment records. It must also contain *FSHIRE* as the approval category code. ■

Workflow Model

The following model illustrates the Hire a Full Time Staff Member workflow. The circled numbers refer to the processing details that follow.



Processing Details

The automated activity, `GET_EMPLOYEE_INFO`, retrieves employee information (1.0). The automated activity, `GET_PAF_INFO`, retrieves PAF information (2.0).

The automated activity, `Get_Citizen_Code`, retrieves the citizen code (3.0). If there is no citizen code, an interactive activity displays the Identification Form (PPAIDEN) so the Employment Manager can enter the citizenship information (3.1). After the information has been entered and saved, the citizen code is retrieved, and the workflow continues.

If the new hire is not a U.S. citizen (`SPBPERS_CITZ_CODE = "N"`), an automated activity, `Get_Visa_Code`, retrieves the visa code (4.0). A business rule checks to see whether the current visa code is valid for working in the U.S. (`SPRINTL_VTYP_CODE_CURRENT = "F1", "B1", "J1", "PR", or "RA"`).

- If the value for the visa code is null, an interactive activity, `Verify_and_Enter_Current_Visa`, displays the International Information Form (GOAINTL) so the Employment Manager can add it (5.0). Then the workflow retrieves the current visa code (4.0), and the workflow continues.
- If the visa code is invalid, a manual notification, `Notify_Employee_of_Missing_or_Expired_Visa`, tells the Hiring Manager to notify the new hire that he/she cannot work in the U.S. (6.0). A manual activity, `Notify_Initiator_That_Employee_Does_Not_Have_A_Valid_Visa`, tells the Hiring Manager that the hiring process is incomplete and the employment PAF must be voided and resubmitted (7.0). The interactive activity, `Void_PAF`, displays the Electronic Personnel Action Form (NOAEPAF) so the Hiring Manager can void the PAF (8.0). The workflow ends.
- If the visa code is valid, the workflow continues.

If the new hire is a U.S. citizen (`SPBPERS_CITZ_CODE = "Y"`), the workflow sends the Budget Director a notification that someone has been hired to fill the position and asks for verification that the position is funded in the budget. The Budget Director uses the Position Budget Form (NBAPBUD) to verify that there is money in the budget to fund the position (9.0).

An automated activity, `Get_Next_PAF_Approver`, looks for an approver (10.0).

- If no approver is found, a manual activity, `Notify_Initiator_No_Approver_Userid_Found`, notifies the Hiring Manager that no approver was found and that the new hire needs to be applied manually (11.0). The workflow ends.
- If an approver is found, the interactive activity, `Approve_Staff_PAF`, sends the NOAEPAF Form to the designated approver (12.0).

The designated approver enters information on the NOAEPAF Form. The next activity depends on the resulting status of the NOAEPAF Form:

- If the PAF is pending, an automated activity, `Get_Next_PAF_Approver`, looks for the next approver (10.0), and the workflow continues.
- If the PAF is returned for correction, a directed activity, `Notify_Initiator_that_PAF_was_Returned_for_Correction`, notifies the Hiring Manager (13.0), and the workflow ends.
- If the PAF is canceled, a directed activity, `Notify_Initiator_that_the_PAF_has_Been_Cancelled`, notifies the Hiring Manager (14.0), and the workflow ends.
- If the PAF is disapproved, a directed activity, `Notify_Initiator_that_PAF_was_Disapproved`, notifies the Hiring Manager (15.0), and the workflow ends.
- If the PAF is voided, a directed activity, `Notify_Initiator_that_PAF_Has_Been_Voided`, notifies the Hiring Manager (16.0), and the workflow ends.
- If the PAF is approved, partially completed, or completed, the workflow continues.

The directed activity, `Notify_to_Apply_PAF`, sends an e-mail to the Hire Originator that the PAF is ready to be applied (17.0). The Hiring Manager notifies the individual responsible for applying staff PAF's to apply the transaction in the Banner database.

An automated activity, `Get_PAF_Status`, retrieves the PAF status (18.0):

- If the PAF is not complete, the directed activity, `Notify_that_PAF_has_not_been_Applied_Successfully`, sends a manual notification to the Hiring Manager that the PAF has not been applied successfully, and the workflow ends (19.0). The originator must create a new transaction from the beginning.
- If the PAF is complete, the following activities are triggered simultaneously:
 - An e-mail activity, `Notify_HR_of_New_Staff_Hire`, notifies the Employment Manager of the new staff member (20.0).
 - An e-mail activity, `Notify_Benefits_Office_to_Establish_Enrollment`, notifies the Benefits Administrator of the new staff member (21.0).
 - An e-mail activity, `Notify_Payroll_Office_For_Enrollment_Of_Taxes_And_Direct_Deposits`, notifies the Payroll Manager of the new staff member (22.0).
 - An e-mail activity, `Notify_Orientation_Coordinator_Of_New_Staff_Hire`, notifies the Orientation Coordinator of the new staff member (23.0).

Database Objects

The Hire a Full Time Staff Member workflow uses the following database objects:

Triggers				
Name	Description/Where Used	New	Creation script	Dependencies
NT_WFAUR_PAFCHANGE_UPDATE	Inserts a row into event queue when status on PAF status table NOBTRAN changes to <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN0.SQL	
NT_WFAIR_PAFCHANGE_INSERT	Inserts a row into event queue when a new row is inserted on PAF status table and its status is <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN1.SQL	

Functions				
Name	Description/Where Used	New	Creation script	Dependencies
F_GET_NOBTRAN_STATUS	Determines the PAF's status after the notification to apply the PAF. Depending on the PAF's status, the workflow branches to completion or continues.	Yes	NOKMISC.SQL	

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
POKWLIB.P_GET_EMPL_INFO	Gets the person's name, employee class, and home department.	Yes	POKWLIB.SQL POKWLI1.SQL	Payroll must be present.
POKWLIB.P_GET_PAF_POSN_INFO	Gets the PAF position description, originator userid, and originator name.	Yes	POKWLIB.SQL POKWLI1.SQL	Payroll must be present.

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
P_NOBTRANPAF CHANGE	Package called by the above trigger to insert the appropriate data into the Event Queue tables.	Yes	NOKWEVN.SQL NOKWEV1.SQL	
P_GET_NEXT_PAF_ USERID	Gets the next userid that needs to approve the PAF and information on whether this ID was obtained. Fired by the automated activity if the ID is eligible for hiring.	Yes	NOKWLIB.SQL NOKWLI1.SQL	
P_GET_CITIZEN_CODE	Gets the citizen code.	Yes	POKWLIB.SQL POKWLI1.SQL	
P_GET_VISA_CODE	Gets the visa code.	Yes	POKWLIB.SQL POKWLI1.SQL	

Establish Benefits

This workflow:

- Notifies the Benefits Administrator to send benefits enrollment forms to the new hire.
- Sets up the new hire's union membership, if necessary.
- Sets up relocation benefits.
- Records beneficiary information and coverages.
- Records benefit elections.
- Instructs the Benefits Administrator to generate a verification of benefits for the new hire.

Roles

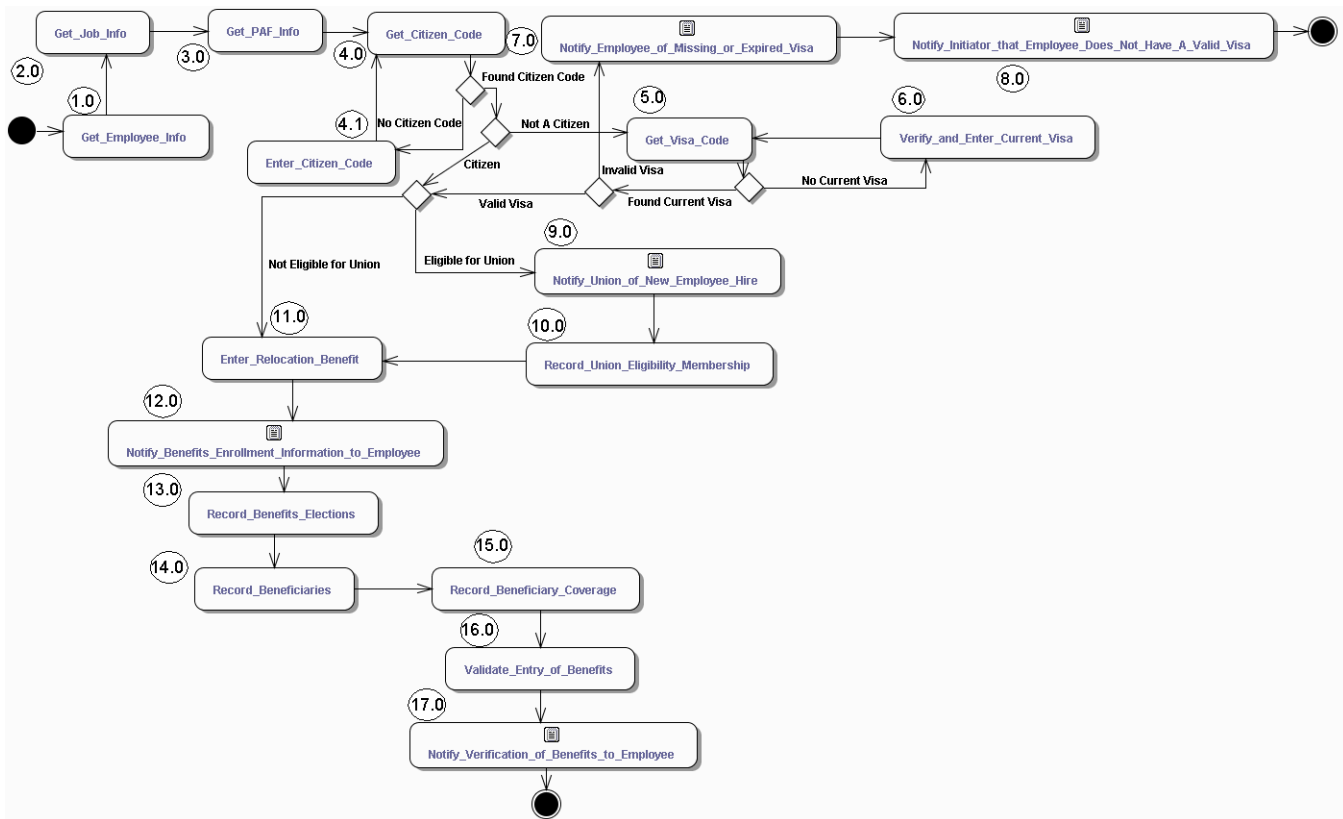
Role	User ID
Employment Manager	hrisusr
Benefits Administrator	hrisusr

Workflow Initiation

This workflow is initiated automatically when the corresponding Hire a Full Time Faculty Member workflow or Hire a Full Time Staff Member workflow finishes. This workflow can run at the same time as the Load Faculty Information workflow and the Establish Taxes and Other Deductions workflow.

Workflow Model

The following model illustrates the Establish Benefits workflow. The circled numbers refer to the processing details that follow.



Processing Details

An automated activity, `Get_Employee_Info`, queries the biographical record from the Identification Form (PPAIDEN) to determine whether the new hire requires specialized tax records (1.0).

An automated activity, `Get_Job_Info`, retrieves job information (2.0). An automated activity, `Get_PAF_Info`, retrieves PAF information (3.0).

The automated activity, `Get_Citizen_Code`, retrieves the citizen code (4.0). If there is no citizen code, an interactive activity displays the Identification Form (PPAIDEN) so the Employment Manager can enter the citizenship information (4.1). After the information is entered and saved, the citizen code is retrieved, and the workflow continues.

If the new hire is a U.S. citizen (`SPBPERS_CITZ_CODE = "Y"`), the workflow continues.

If the new hire is not a U.S. citizen (`SPBPERS_CITZ_CODE = "N"`), an automated activity, `Get_Visa_Code`, retrieves the visa code (5.0). A business rule checks to see whether the current visa code is valid for working in the U.S. (`SPRINTL_VTYP_CODE_CURRENT = "F1", "B1", "J1", "PR", or "RA"`).

- If the value for the visa code is null, an interactive activity, `Verify_and_Enter_Current_Visa`, displays the International Information Form (GOAINTL) so the Employment Manager can add it (6.0). Then the workflow retrieves the current visa code (5.0), and the workflow continues.
- If the visa code is invalid, a manual notification, `Notify_Employee_of_Missing_or_Expired_Visa`, tells the Employment Manager to notify the new hire that he/she cannot work in the U.S. (7.0). A manual activity, `Notify_Initiator_That_Employee_Does_Not_Have_A_Valid_Visa`, tells the Hire Originator that the hiring process is incomplete and the employment PAF must be voided and resubmitted (8.0). The PAF is not voided, but the workflow ends.
- If the visa code is valid, the workflow continues.

If the new hire is eligible for union membership, based on the employee class code (`PEBEMPL_ECLS_CODE`), two activities occur:

- A manual activity, `Notify_Union_of_New_Employee_Hire`, tells the Employment Manager to notify the bargaining unit or union that the new hire is eligible for union membership (9.0).
- An interactive activity, `Record_Union_Eligibility_Membership`, displays the Employee/Job Labor Relations Form (PEABARG) to the Employment Manager so union information can be entered for the new hire (10.0).

An interactive activity, `Enter_Relocation_Benefit`, displays the Employee Jobs Form (NBAJOBS) so the Benefits Administrator can enter relocation benefits (11.0). Relocation benefits can be entered as a lump sum or spread out over time, depending on the tax implications. When NBAJOBS appears, it is prefilled with the ID, position code, and suffix code.

A manual activity, `Notify_Benefits_Enrollment_Information_to_Employee`, tells the Benefits Administrator to send benefits enrollment information to the new hire (12.0). This manual activity is completed when the forms are returned.

When the forms are returned, the interactive activity, `Record_Benefits_Elections`, displays the Employee Benefit/Deduction Setup Form (PDABDSU) (13.0). The Benefits Administrator enters the benefit elections.

An interactive activity, `Record_Beneficiaries`, displays the Beneficiary Form (PDABENE) so the Benefits Administrator can record beneficiary information for the new hire (14.0).

An interactive activity, `Record_Beneficiary_Coverage`, displays the Beneficiary Coverage Form (PDABCOV) so the Benefits Administrator can enter coverage information for the beneficiaries of the new hire (15.0).

An interactive activity, `Validate_Entry_of_Benefits`, displays the benefits enrollment options on the Employee Deduction Query Form (PDIDLST). The Benefits Administrator can visually check the information entered onto the new hire's record (16.0).

A manual activity, `Notify_Verification_of_Benefits_to_Employee`, notifies the Benefits Administrator to generate a Verification of Benefits for the new hire (17.0). When the Benefits Administrator acknowledges this activity, the workflow ends.

Database Objects

The Establish Benefits workflow uses the following database objects:

Triggers				
Name	Description/Where Used	New	Creation script	Dependencies
NT_WFAUR_PAFCHANGE_UPDATE	Inserts a row into event queue when status on PAF status table NOBTRAN changes to <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN0.SQL	
NT_WFAIR_PAFCHANGE_INSERT	Inserts a row into event queue when a new row is inserted on PAF status table and its status is <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN1.SQL	
Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
POKWLIB.P_GET_EMPL_INFO	Gets the person's name, employee class, and home department.	Yes	POKWLIB.SQL POKWLI1.SQL	Payroll must be present.
POKWLIB.P_GET_PAF_POSN_INFO	Gets the PAF position description, originator userid, and originator name.	Yes	POKWLIB.SQL POKWLI1.SQL	Payroll must be present.
P_NOBTRANPAFCHANGE	Package called by the above triggers to insert the appropriate data into the Event Queue tables.	Yes	NOKWEVN.SQL NOKWEV1.SQL	
POKWLIB.P_GET_JOBS_INFO	Gets the job title, annual salary, and timesheet orgn.	Yes	POKWLIB.SQL POKWLI1.SQL	Payroll must be present.
P_GET_CITIZEN_CODE	Gets the citizen code.	Yes	POKWLIB.SQL POKWLI1.SQL	
P_GET_VISA_CODE	Gets the visa code.	Yes	POKWLIB.SQL POKWLI1.SQL	

Load Faculty Information

The Academic Affairs office must load information about a faculty member into Banner whenever a new faculty member is hired.

Roles

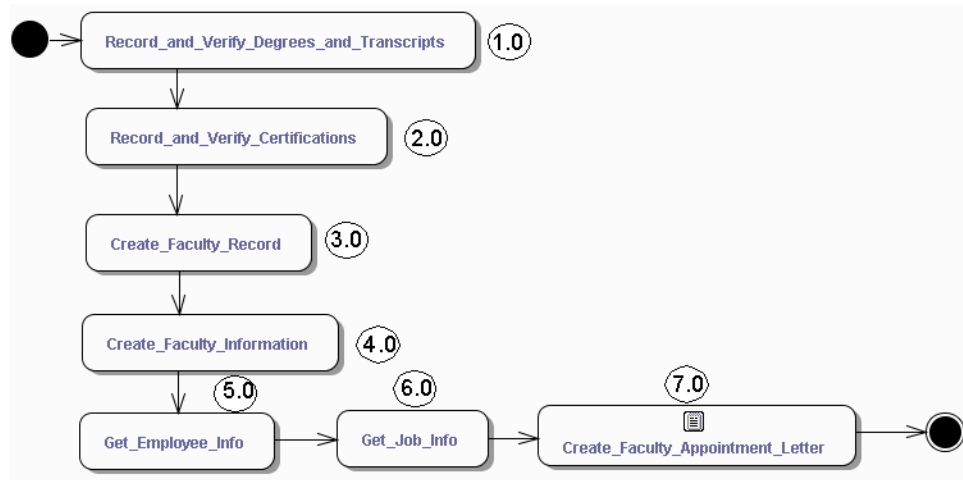
Role	User ID
Academic Affairs Administrator	genlprd

Workflow Initiation

This workflow is initiated automatically when the corresponding Hire a Full Time Faculty Member workflow finishes. It can run at the same time as the Establish Benefits workflow and the Establish Taxes and Other Deductions workflow.

Workflow Model

The following model illustrates the Load Faculty Information workflow. The circled numbers refer to the processing details that follow.



Processing Details

An interactive activity, `Record_and_Verify_Degrees_and_Transcripts`, displays the General Information Form (PPAGENL) for the Academic Affairs Administrator (1.0). The workflow passes the new hire's ID into the form. The Academic Affairs manager enters the following information:

- Institutions where the new faculty member previously worked
- Degrees the new faculty member earned
- Transcript information for the new faculty member

An interactive activity, `Record_and_Verify_Certifications`, displays the Certification Information Form (PPACERT) for the Academic Affairs Administrator (2.0). The workflow passes the new hire's ID into the form. The Academic Affairs Administrator enters the following information:

- Certifications the new faculty member earned
- Dates when the certifications expire
- Agencies that endorse the certifications

An interactive activity, `Create_Faculty_Record`, displays the Faculty Action Tracking Form (PEAFAC) for the Academic Affairs Administrator (3.0). The workflow passes the new hire's ID into the form, and the system checks the faculty indicator on the Installation Rule Form (PTRINST). The form advances to the General Information window. If the query does not retrieve the Begin Date, the workflow automatically enters the Job Effective Date from the new hire's job record. The Academic Affairs Administrator enters the following information on PEAFAC:

- Faculty appointment information
- Tenure information
- Rank and review data
- Faculty sabbaticals and leaves

An interactive activity, `Create_Faculty_Information`, displays the Faculty Information Form (SIAINST) for the Academic Affairs Administrator, who enters faculty information (4.0).

An automated activity, `Get_Employee_Info`, gets employee information (5.0). An automated activity, `Get_Job_Info`, gets job information (6.0)

A manual activity, `Create_Faculty_Appointment_Letter`, notifies the Academic Affairs Administrator that faculty information is loaded (7.0). A faculty appointment letter should be prepared and sent to the employee for signature.

Database Objects

The Establish Benefits workflow uses the following database objects:

Triggers				
Name	Description/Where Used	New	Creation script	Dependencies
NT_WFAUR_PAFCHANGE_UPDATE	Inserts a row into event queue when status on PAF status table NOBTRAN changes to <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN0.SQL	
NT_WFAIR_PAFCHANGE_INSERT	Inserts a row into event queue when a new row is inserted on PAF status table and its status is <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN1.SQL	
Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
P_NOBTRANPAFCHANGE	Package called by the above triggers to insert the appropriate data into the Event Queue tables.	Yes	NOKWEVN.SQL NOKWEV1.SQL	
POKWLIB.P_GET_JOBS_INFO	Gets the job title, annual salary, and timesheet orgn.	Yes	POKWLIB.SQL POKWL11.SQL	Payroll must be present.

Establish Taxes and Other Deductions

This workflow helps the Employment Manager and Payroll Clerk set up taxes and deductions for new full-time faculty and staff members in Banner.

Roles

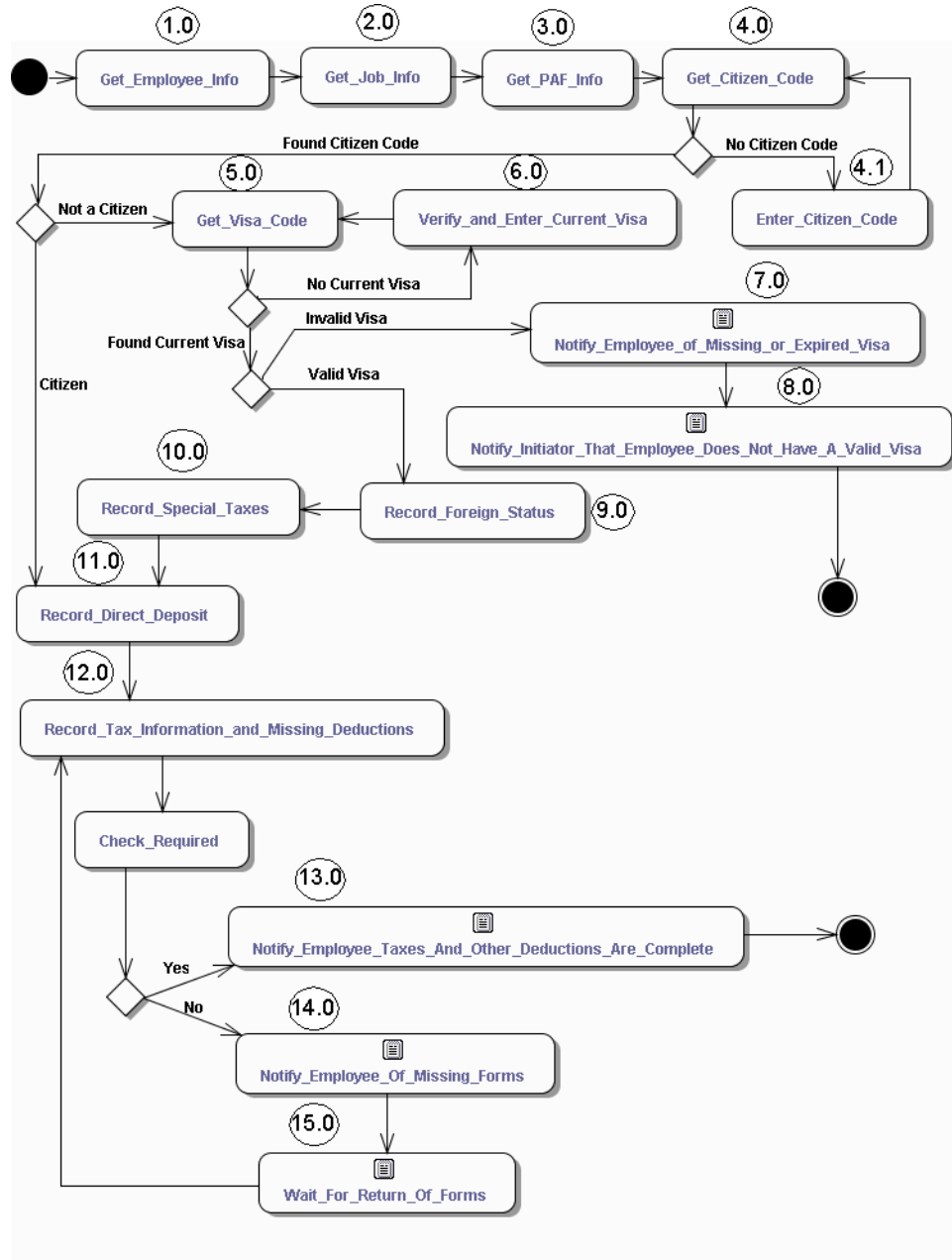
Role	User ID
Payroll Clerk	hrisusr
Employment Manager	hrisusr

Workflow Initiation

This workflow is initiated automatically when the corresponding Hire a Full Time Faculty Member workflow or the Hire a Full Time Staff Member workflow finishes. It can run at the same time as the Establish Benefits workflow and the Load Faculty Information workflow.

Workflow Model

The following model illustrates the Establish Taxes and Other Deductions workflow. The circled numbers refer to the processing details that follow.



Processing Details

An automated activity, `Get_Employee_Info`, queries the biographical record from the Identification Form (PPAIDEN) to determine whether the new hire requires specialized tax records (1.0).

An automated activity, `Get_Job_Info`, retrieves job information (2.0). An automated activity, `Get_PAF_Info`, gets PAF information (3.0).

The automated activity, `Get_Citizen_Code`, retrieves the citizen code (4.0). If there is no citizen code, an interactive activity, `Enter_Citizen_Code`, displays the Identification Form (PPAIDEN) so the Employment Manager can enter citizenship information (4.1). After the information is entered and saved, the citizen code is retrieved, and the workflow continues.

If the new hire is a U.S. citizen (`SPBPERS_CITZ_CODE = "Y"`), the workflow continues.

If the new hire is not a U.S. citizen (`SPBPERS_CITZ_CODE = "N"`), an automated activity, `Get_Visa_Code`, retrieves the visa code (5.0). A business rule checks to see whether the current visa code is valid for working in the U.S. (`SPRINTL_VTYP_CODE_CURRENT = "F1", "B1", "J1", "PR", or "RA"`).

- If the value for the visa code is null, an interactive activity, `Verify_and_Enter_Current_Visa`, displays the International Information Form (GOAINTL) so the Employment Manager can add it (6.0). Then the workflow retrieves the current visa code (5.0), and the workflow continues.
- If the visa code is invalid, a manual notification, `Notify_Employee_of_Missing_or_Expired_Visa`, tells the Employment Manager to notify the new hire that he/she cannot work in the U.S. (7.0). A manual activity, `Notify_Initiator_That_Employee_Does_Not_Have_A_Valid_Visa`, tells the Hire Originator that the hiring process is incomplete and the employment PAF must be voided and resubmitted (8.0). The PAF is not voided, but the workflow ends.
- If the visa code is valid, the workflow continues.

In addition, the following activities are performed for foreign new hires:

- The interactive activity, `Record_Foreign_Status`, displays the Employee Form (PEAEMPL). The Employment Manager enters the foreign tax status in the Regulatory Information window (9.0).
- The interactive activity, `Record_Special_Taxes`, displays the Employee Benefit/Deduction Set Up Form (PDABDSU). The Payroll Clerk enters special taxes (10.0).

An interactive activity, `Record_Direct_Deposit`, displays the Direct Deposit Recipient Form (GXADIRD). The Payroll Clerk records all the direct deposit information for the new hire, as necessary (11.0).

An interactive activity, `Record_Tax_Information_and_Missing_Deductions`, displays the Employee Benefit/Deduction Setup Form (PDABDSU). The Payroll Clerk enters other taxes and any other deductions for the new hire, as necessary (12.0).

If all the required deductions have been entered, a manual activity, `Notify_Employee_Taxes_And_Other_Deductions_Are_Complete`, notifies the Payroll Clerk that enrollment of all the required payroll taxes and other deductions is complete (13.0). Once the employee is notified that enrollment is complete, the workflow ends.

If all the required deductions have not been entered, a manual activity, `Notify_Employee_Of_Missing_Forms`, notifies the Payroll Clerk that some benefits and/or deductions information needed to run Payroll is missing (14.0). The employee must be notified to provide the necessary information. When the information is returned (15.0), the Payroll Clerk acknowledges its receipt within Banner Workflow. The Employee Benefit/Deduction Setup Form (PDABDSU) appears (12.0).

Database Objects

The Establish Taxes and Other Deductions workflow uses the following database objects:

Triggers				
Name	Description/Where Used	New	Creation script	Dependencies
NT_WFAUR_PAFCHANGE_UPDATE	Inserts a row into event queue when status on PAF status table NOBTRAN changes to <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN0.SQL	
NT_WFAIR_PAFCHANGE_INSERT	Inserts a row into event queue when a new row is inserted on PAF status table and its status is <i>P</i> (pending) or <i>C</i> (completed).	Yes	NOTPTRN1.SQL	

Functions				
Name	Description/Where Used	New	Creation script	Dependencies
F_CHECK_REQD_DEDNS	Checks whether all required deductions have been set up. Located in an activity in a workflow that fires after taxes are set up.	Yes	POKLIBS.SQL POKLIB1.SQL	Payroll must be present.

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
POKWLIB.P_GET_EMPL_INFO	Gets the person's name, employee class, and home department.	Yes	POKWLIB.SQL POKWL11.SQL	Payroll must be present.
POKWLIB.P_GET_PAF_POSN_INFO	Gets the PAF position description, originator userid, and originator name.	Yes	POKWLIB.SQL POKWL11.SQL	Payroll must be present.
P_NOBTRANPAFCHANGE	Package called by the above triggers to insert the appropriate data into the Event Queue tables.	Yes	NOKWEVN.SQL NOKWEV1.SQL	
POKWLIB.P_GET_JOBS_INFO	Gets the job title, annual salary, and timesheet orgn.	Yes	POKWLIB.SQL POKWL11.SQL	Payroll must be present.
P_GET_CITIZEN_CODE	Gets the citizen code.	Yes	POKWLIB.SQL POKWL11.SQL	
P_GET_VISA_CODE	Gets the visa code.	Yes	POKWLIB.SQL POKWL11.SQL	

New Employee Security Setup

A hiring manager (with the role Hire Originator) can use this workflow to begin the process of requesting that a new employee hire be given security access throughout Banner. This workflow involves creating an Oracle username, associating the username with the new hire's Banner ID, establishing appropriate access to Banner Finance and Human Resources (if necessary) and informing the hiring manager via email when all steps have been completed.

Roles

Role	User ID
Hire Originator	genlprd
Database Administrator	genlprd
Finance Security Administrator	fimsusr
Human Resources Security Administrator	hrisusr

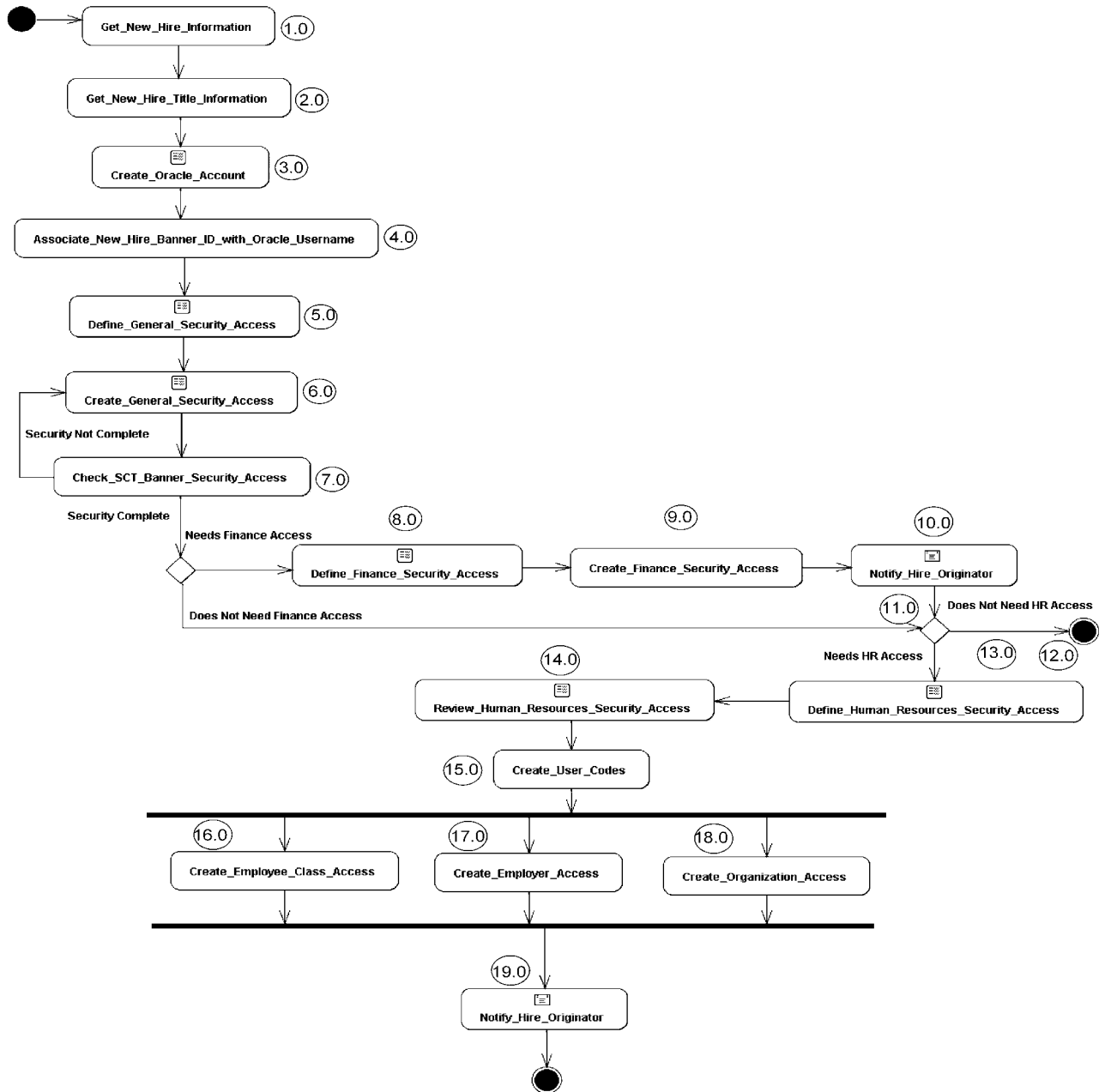
Workflow Initiation

A new hire in the Human Resources database initiates this workflow if both of the following conditions are met for the PAFCHANGE business event:

- The status of the online PAF is Complete.
- The approval category is FFHIRE (full-time faculty) or FSHIRE (full-time staff).

Workflow Model

The following model illustrates the New Employee Security Access workflow. The circled numbers refer to the processing details that follow.



Processing Details

An automated activity, GET_NEW_HIRE_INFORMATION, retrieves the name and department of the new hire (1.0). Another automated activity, GET_NEW_HIRE_TITLE_INFORMATION, retrieves the title and other relevant information for the new hire (2.0).

The next step in the process is a custom activity, CREATE_ORACLE_ACCOUNT, that sends a notice to the Database Administrator asking that an Oracle Username be created for the new hire (3.0).

An interactive activity, ASSOCIATE_NEW_HIRE_BANNER_ID_WITH_ORACLE_USERNAME, displays the Enterprise Access Control Form (GOAEACC) and allows the Database Administrator to build an association between the newly created Banner ID and Oracle Username (4.0).

A custom activity, DEFINE_GENERAL_SECURITY_ACCESS, is then sent to the Hire Originator and prompts them for the type of access to Banner that the new hire will require. The Hire Originator is then able to specify if the new hire will need a Web PIN, Banner Roles, Banner Security Classes, and Departmental Level Access (Finance and/or Human Resources) (5.0).

A custom activity, CREATE_GENERAL_SECURITY_ACCESS (6.0), is then sent to the Database Administrator to establish the appropriated Banner General privileges.

The next step in the process, CHECK_SCT_BANNER_SECURITY_ACCESS (7.0), verifies that the new hire is a valid user on the Security tables.

Following the verification, the business process evaluates a business rule to determine whether or not the Hire Originator requested Banner Finance access for the new hire. If the new hire requires access to Finance, a custom activity, DEFINE_FINANCE_SECURITY_ACCESS (8.0) is sent to the Hire Originator for further definition. When access requirements have been defined, an interactive activity, CREATE_FINANCE_SECURITY_ACCESS (9.0) is routed to the Finance Security Administrator for establishment and an e-mail notification is sent back to the Hire Originator (10.0) when the setup is complete.

In step (11.0) another business rule is used to ascertain whether or not Human Resources privileges are necessary. If they are not the workflow process ends at step (12.0). If HR is required, a custom activity, DEFINE_HUMAN_RESOURCES_SECURITY_ACCESS (13.0) is sent to the Hire Originator for further definition. When access requirements have been defined, a custom activity, REVIEW_HUMAN_RESOURCES_SECURITY_ACCESS is sent to the HR Security Administrator for approval (14.0).

Following approval of the HR security, a series of automated activities are launched to update the appropriated tables in Banner Human Resources. Automated activities CREATE_USER_CODES (15.0), CREATE_EMPLOYEE_CLASS_ACCESS (16.0), CREATE_EMPLOYER_ACCESS (17.0), and CREATE_ORGANIZATION_ACCESS (18.0) update the HR tables.

An e-mail activity, NOTIFY_HIRE_ORIGINATOR (19.0) sends a message back to the Hire Originator to let them know that all the setup has been completed.

Database Objects

The New Employee Security Setup workflow uses the following database objects:

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
POKWLIB.P_GET_PAF_EMPL_INFO	Gets the name of the new hire, Employee Class, Organization, and Department	No	POKWLIL.SQL	
POKWLIB.P_GET_PAF_POSN_INFO	Gets the title of the new hire, and the name and user ID of the Hire Originator	No	POKWLIL.SQL	
P_CHECK_BANNER_SECURITY_ACCESS	Returns True or False, depending on whether the user is a valid user in the Security table.	No		
P_CREATE_USER_CODE_RULES	Updates the HR User Tables with Master Employee Class, Employer, Organization, and Max Salary for the new hire.	Yes		
P_CREATE_EMPL_CLASS_SECURITY	Defines Employee Class access for the new hire	Yes		

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
P_CREATE_EMPLOYER_SECURITY	Defines Employer Code for the new hire.	Yes		
P_CREATE_ORGINIZATION_SECURITY	Defines the Chart of Accounts, Organization High, and Organization Low amounts for the new hire.	Yes		

Withdraw a Student

This workflow lets an institution withdraw a student from course work for a given term or terms, then informs various departments within the institution. Each department performs a separate process. For example, if the student received financial aid, the Process Withdrawn Student Aid workflow is initiated. The informed departments include:

- Housing
- Veteran's Affairs
- International Affairs
- Bursar's Office
- Registrar's Office

Roles

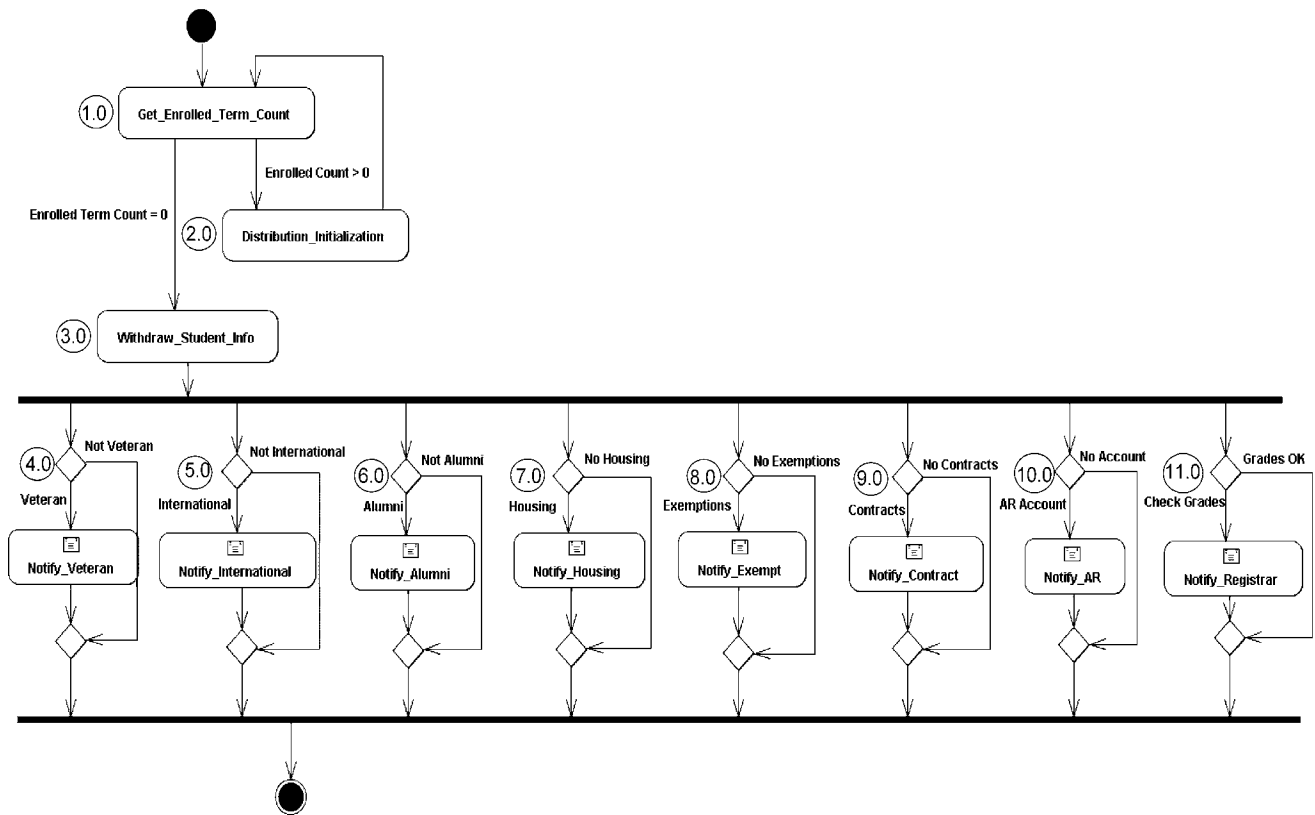
Role	User ID
Registrar	saisusr
Veterans Affairs	saisusr
International Affairs	saisusr
Development Officer	adisusr
Student Housing	saisusr
Bursar Office Clerk	fimsusr

Workflow Initiation

The student initiates the withdrawal by completing and taking a form to the administrative office that processes withdrawals. When the office processes the form, a withdrawal reason is added to the Term Course Maintenance Form (SHAINST). When that record is saved, the `WITHDRAWSTUDENT`, business event is fired. This business event initiates the workflow.

Workflow Model

The following model illustrates the Withdraw a Student workflow. The circled numbers refer to the processing details that follow.



Processing Details

An automated activity, `GetEnrolledTermCount`, determines the student's enrolled term count (1.0).

If the student is registered for courses for the same term as when the withdrawal record was entered, or for any future courses, the interactive activity, `DistributionInitialization`, displays the Student Course Registration Form (SFAREGS) for the Registrar (2.0). The Registrar removes the student from all courses and performs the online fee assessment. After the courses are removed, the automated activity, `GetEnrolledTermCount`, checks again to make sure the enrolled term count equals zero.

If the enrolled term count is zero, the automated activity, `WithdrawStudentInfo`, retrieves additional student information and initiates the Process Withdrawn Student Aid workflow (3.0).

The following activities occur simultaneously:

- If the student has a record on the Veteran Information window of the General Student Form (SGASTDN) for the term(s) for which the student is withdrawing, the Veterans Affairs Office receives an e-mail, `NotifyVeteran`, to review the student's veteran certification records (4.0).
- If the student has a visa type code that is defined on the Visa Type Code Validation Form (STVVVYP) with the **Non Res** indicator checked (the student is not a resident of the United States), the International Affairs Office receives an e-mail, `NotifyInternational`, to review the student's international records (5.0).
- If the student is an alumnus, the Development Office receives an e-mail, `NotifyAlumnus`, to review the student's records (6.0).
- If the student has a housing record on the Room Assignment Form (SLARASG) for the term(s) from which the student is withdrawing, the Student Housing Office receives an e-mail, `NotifyHousing`, to review the student's room, meal, and telephone assignments (7.0).
- If the student has an exemption record for the term(s) defined on the Exemption Authorization Form (TSAEXPT), the Bursar Office Clerk receives an e-mail, `NotifyExempt`, to review the student's exemptions for the term (8.0).
- If the student has a third party contract for the term(s) defined on the Contract Authorization Form (TSACONT), the Bursar Office Clerk receives an e-mail, `NotifyContract`, to review the student's contracts for the term (9.0).
- If the student has accounts receivable, the Bursar Office Clerk receives an e-mail, `NotifyAR`, to review the student's account (10.0).
- If the course registration statuses on the courses withdrawn do not have a final grade, the Registrar receives a manual activity, `NotifyRegistrar`, to review the student's registration records (11.0).

Database Objects

The Withdraw a Student workflow uses the following database objects:

Triggers				
Name	Description/Where Used	New	Creation script	Dependencies
ST_WFAIR_WITHDRAW_STUDENT	Initiates the event on an insert where the Withdraw Reason (SHRTTRM_WRSN_CODE) is not null.	Yes	SHTRTM1.SQL	
ST_WFAUR_WITHDRAW_STUDENT	Initiates the event on an insert where the Withdraw Reason (SHRTTRM_WRSN_CODE) is not null and has been modified.	Yes	SHTRTM2.SQL	

Procedures				
Name	Description/Where Used	New	Creation script	Dependencies
SHKWEVN.P_WITHDRAW_STUDENT	Updates row triggers on SHRTTRM.	No	SHKWEVN.SQL SHKWEV1.SQL	
SHKWEVN.P_WITHDRAW_STUDENT	Returns data to control subsequent Banner Workflow tasks. Initiates the Process Withdrawn Student Aid workflow.	No	SHKWEVN.SQL SHKWEV1.SQL	

Process Withdrawn Student Aid

This workflow supports the Withdraw a Student workflow. It is automatically triggered if a withdrawn student receives financial aid for the term(s) or aid year from which the student is withdrawing. The workflow recalculates, adjusts, and cancels awards, Title IV loans, and work study benefits.

This example workflow has some important exceptions:

- This example workflow captures only those students who have aid awarded within the aid year of their withdrawal. This workflow does not capture a student who received a loan in a prior year and has not been awarded any aid in the current year.
- This example workflow does not identify students who received loans that were processed exclusively through the Loan module and were never posted as awards to the Award Form (RPAAWRD).
- The amount paid for a loan is seldom the same as the amount of the offer (Federal Perkins can be the same). A student with loans is flagged as having unprocessed aid and potentially as having outstanding disbursements. Someone must review the loans to determine whether any additional work is required.

Roles

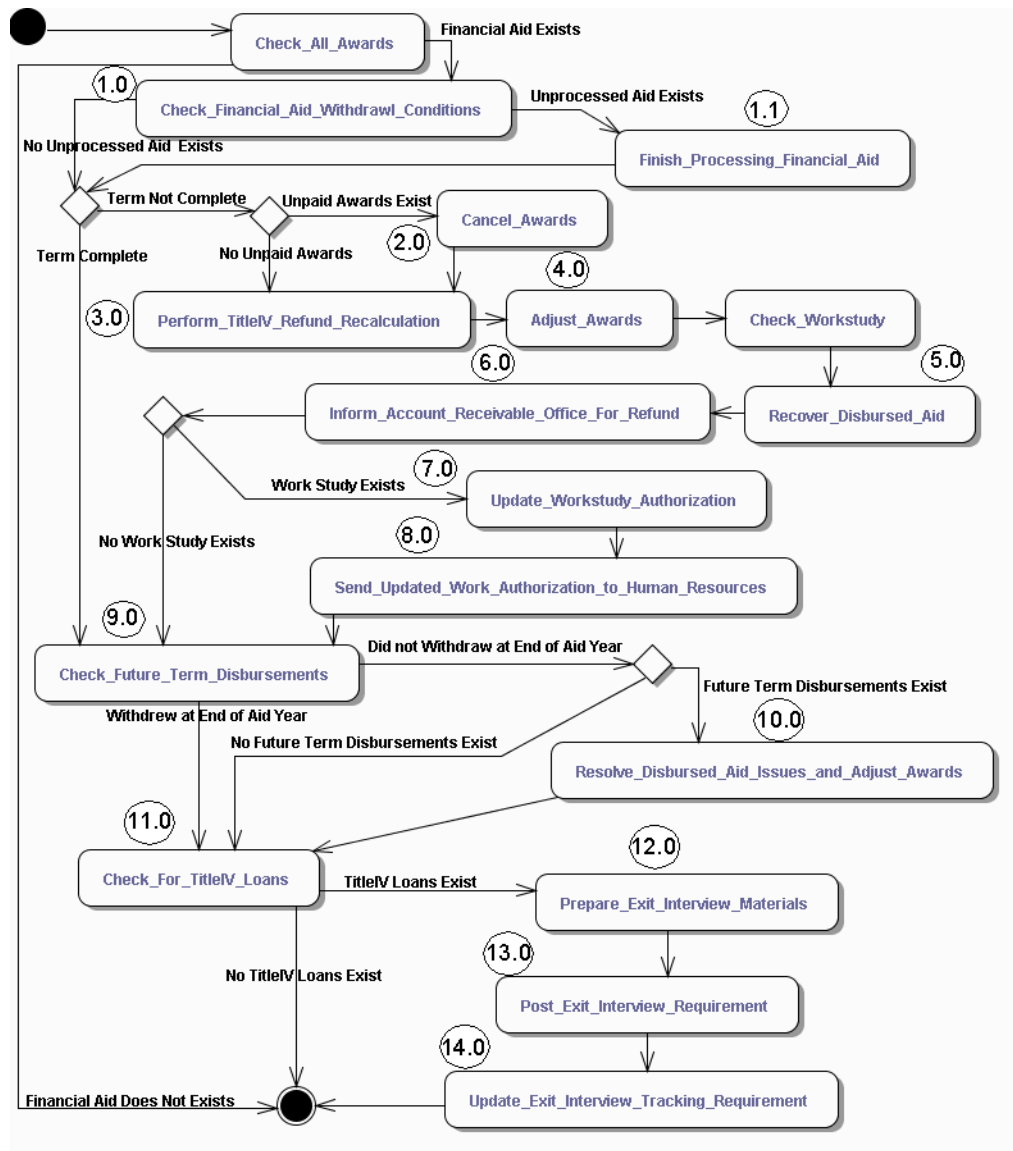
Role	User ID
Loan Processor	faisusr
Work Study Coordinator	faisusr
Withdrawal Coordinator	faisusr
Payroll Clerk	fimsusr
Bursar Office Clerk	fimsusr

Workflow Initiation

The business event `FAWITHDRAW`, which is part of the Withdraw a Student workflow, triggers this workflow if the student has financial aid at the institution for the current aid year.

Workflow Model

The following model illustrates the Process Withdrawn Student Aid workflow. The circled numbers refer to the processing details that follow.



Processing Details

An automated activity, `Check_Financial_Aid-Withdrawal_Conditions` (1.0), determines whether:

- Any unprocessed aid exists for the student in this aid year.
- The student withdrew before the current term was completed.
- Any unpaid awards exists for the student in this aid year.

If unprocessed aid exists, an interactive activity, `Finish_Processing_Financial_Aid`, displays the Award Form (RPAAWRD) on the worklist of the Withdrawal Coordinator (1.1). The Withdrawal Coordinator finishes processing and disbursing financial aid as appropriate for the withdrawn student.

If the student withdrew before the current term was completed and unpaid awards exist, an interactive activity, `Cancel_Awards`, displays the Award Form (RPAAWRD) on the worklist of the Withdrawal Coordinator (2.0). The Withdrawal Coordinator cancels awards that had no activity for the current aid year. Once the awards are canceled, processing continues with the `Perform_TitleIV_Refund_Recalculation` activity (3.0).

If the student withdrew before the current term was completed and no unpaid awards exist, the following activities occur:

- The Withdrawal Coordinator receives a manual notification, `Perform_TitleIV_Refund_Recalculation`, to complete Title IV refund/recalculation processing for the specified student and aid year (3.0).
- An interactive activity, `Adjust_Awards`, displays the Award Form (RPAAWRD) for the Withdrawal Coordinator to adjust records as necessary (4.0).
- An interactive activity, `Recover_Disbursed_Aid`, displays the Applicant Immediate Process Form (ROAIMMP) for the Withdrawal Coordinator to recalculate and recover any aid (5.0).
- The Bursar Office Clerk receives a manual notification, `Inform_Account_Receivable_Office_For_Refund`, to query the account for the specified student and process any refund necessary (6.0).
- If the student was enrolled in a work study program, an interactive activity, `Update_Workstudy_Authorization`, displays the Student Employment Authorization Form (RJASEAR) for the Work Study Coordinator to update the work study awards as necessary (7.0). The Payroll Clerk receives a manual notification, `Send_Updated_Work_Authorization_to_Human_Resources`, to update the records for the specified student (8.0).

Next, an automated activity, `Check_Future_Term_Disbursements`, checks to see whether the student has aid for future terms (9.0). If the student withdrew before the end of the aid year and disbursements exist for a future term, an interactive activity, `Resolve_Disbursed_Aid_Issues_and_Adjust_Awards`, displays the Award Form (RPAAWRD) for the Withdrawal Coordinator to resolve any disbursed aid issues and adjust awards (10.0). Once these adjustments are made, processing continues with a check for Title IV loans (11.0).

An automated activity, `Check_For_TitleIV_Loans`, determines whether any Title IV loans exist for the student (11.0). If the student has no Title IV loans, the workflow ends. If the student has Title IV loans, processing continues with an exit interview (12.0).

The Loan Processor receives a manual notification, `Prepare_Exit_Interview_Materials`, to prepare the exit interview materials for the student (12.0). Once this activity is completed, an interactive activity, `Post_Exit_Interview_Requirement`, displays the Applicant Requirements Form (RRAAREQ) for the Loan Processor to review financial aid tracking requirements with emphasis on Title IV loans (13.0).

Finally, an interactive activity, `Update_Exit_Interview_Tracking_Requirement`, displays the Applicant Requirements Form (RRAAREQ) for the Loan Processor to update tracking requirements (14.0). Then the workflow ends.

Database Objects

The Process Withdrawn Student Aid workflow uses the following database objects:

Functions				
Name	Description/Where Used	New	Creation script	Dependencies
<code>ROKWFLB.F_CHECK_CURRENT_AIDY_AWARD</code>	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has financial aid for the term code supplied via parameter.	Yes	<code>ROKWFLB.SQL</code>	
<code>ROKWFLB.F_CHECK_ALL_AWARDS</code>	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has financial aid for the term code supplied via parameter.	Yes	<code>ROKWFLB.SQL</code>	
<code>ROKWFLB.F_CHECK_UNPROCESSED_AID</code>	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has unprocessed aid for the term code supplied via parameter.	Yes	<code>ROKWFLB.SQL</code>	

Functions				
Name	Description/Where Used	New	Creation script	Dependencies
ROKWFLB.F_CHECK_UNPAID_AWARDS	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has unpaid awards for the term code supplied via parameter.	Yes	ROKWFLB.SQL	
ROKWFLB.F_CHECK_WORKSTUDY	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has work study aid.	Yes	ROKWFLB.SQL	
ROKWFLB.F_CHECK_FUTURE_TERM_DISBURSED	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has disbursed aid for terms greater than the term code supplied via parameter.	Yes	ROKWFLB.SQL	
ROKWFLB.F_CHECK_TITLEIV_LOANS	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has Title IV loans for the term code supplied via parameter.	Yes	ROKWFLB.SQL	
ROKWFLB.F_CHECK_FOR_EXIT_INTERVIEW	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has satisfied exit interview requirements.	Yes	ROKWFLB.SQL	
ROKWFLB.F_CHECK_END_OF_AID_YEAR	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has withdrawn on the end date of the aid year code supplied via parameter.	Yes	ROKWFLB.SQL	
ROKWFLB.F_CHECK_END_OF_TERM	Returns a value <i>Y</i> or <i>N</i> , indicating whether the student has withdrawn on the end date of the aid year code supplied via parameter.	Yes	ROKWFLB.SQL	

Interview Candidate

A hiring manager can use this workflow to schedule an interview for a prospective candidate. Based on the interview, the supervisor provides an evaluation of the candidate making a recommendation to hire or not to hire. If the recommendation is to hire the candidate, the new person is entered in to the Banner database.

Roles

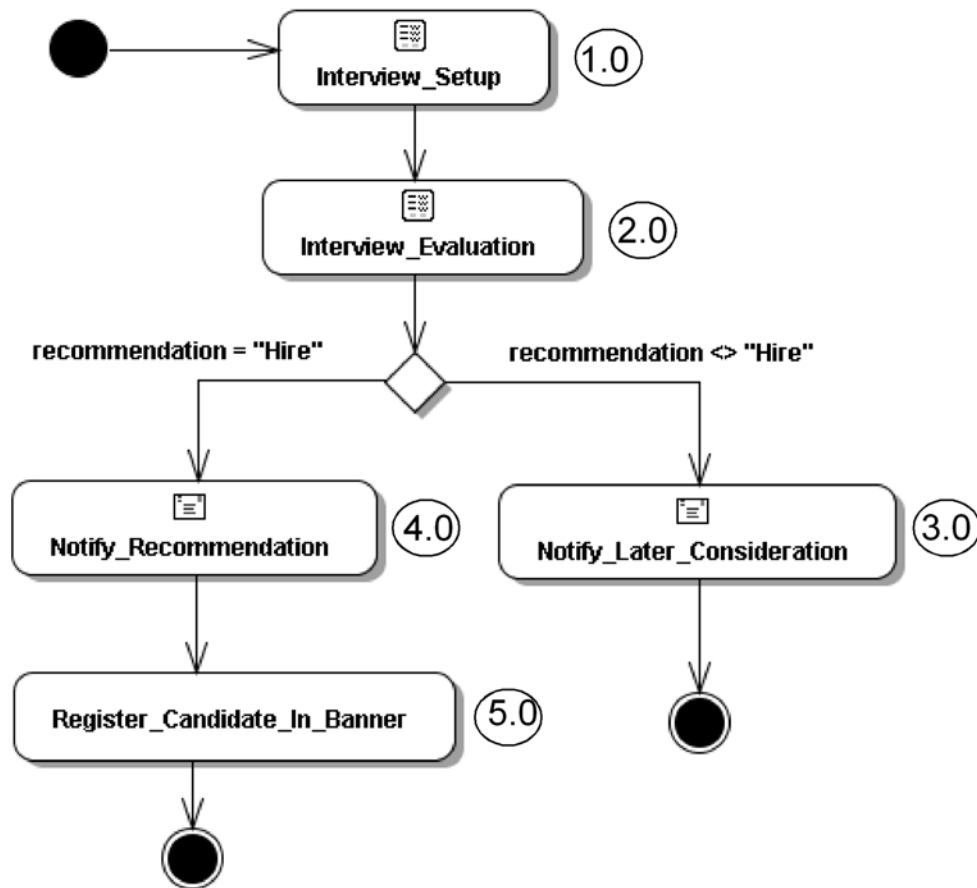
Role	User ID
Hiring Manager	hrisusr
Supervisor	adisusr
	hrisusr
	faisusr
	fimsusr
	saisusr

Workflow Initiation

A hiring manager in Human Resources initiates this workflow by scheduling an interview for a prospective candidate providing details for the candidate, including the interview date, location, position applied and position responsibilities. The hiring manager attaches the candidate's resume for reference.

Workflow Model

The following model illustrates the Interview Candidate workflow. The circled numbers refer to the processing details that follow.



Processing Details

A custom activity, **Interview_Setup**, schedules an interview for a prospective candidate providing the candidate id, last name, first name, interview date and location, position applied for and responsibilities as well as the candidate's resume (1.0).

Another custom activity, **Interview_Evaluation**, sends a notification to the Supervisor with the details of the Upcoming Interview. Once the interview has been completed, the Supervisor evaluates the candidate and makes a recommendation to hire or not to hire based on relevant experience, leadership ability, communication skills and organization skills (2.0).

If the recommendation is not to hire the candidate an email, `Notify_Later_Consideration`, is sent to the Hiring Manager (3.0) and the workflow ends.

If the recommendation is to hire the candidate an email, `Notify_Recommendation`, is sent to the Hiring Manager (4.0). An interactive activity, `Register_Candidate_In_Banner`, displays the SPAIDEN form allowing the hiring manager to create the new person in the Banner database (5.0).

Customizing Example Workflows

You can customize the example workflows delivered by SunGard Higher Education and move them to your production environment.

To successfully customize the delivered examples, you must initially import the same example workflow seed data into your testing and production environments. If you import different data into the two environments, you may need to make additional changes before you can export from the testing environment and import into the production environment. The import will fail if the production environment does not have all the data needed to support the new version of an example workflow.



Note

See Import Banner Components and Examples in Chapter 2 - *Banner Integration of the Workflow Technical Integration Guide*, for a description of seed data and instructions on how to import it. ■

Use the following steps to customize an example workflow:

1. Access Workflow Modeler in the testing environment.
2. Locate the example workflow you wish to modify.
3. Create a new version of the example workflow. Refer to “How to create a new version of a workflow” in chapter 6 of the *Banner Workflow Analyst/Administrator Handbook*.
4. Modify the new version of the example workflow as necessary.



Note

It is important that you keep track of all changes and additions (for example, new and changed components and business events) you make in the testing environment. These changes and additions must be duplicated in the production environment. ■

5. Save, validate, and activate the new version of the example workflow.
6. Run the changed workflow to make sure it works as intended.

7. Selectively export the modified example workflow from the testing environment. Refer to the *Banner Workflow Technical Integration Guide* for instructions.
8. Selectively export any new components and business events that were created in the testing environment and are used in the new version of the example workflow.
9. In the production environment, manually duplicate the changes that were made in the testing environment:
 - Access the BCC in the production environment.
 - Locate each component and business event that was changed in the testing environment.
 - Make the same changes in the production environment.
 - Save the changes.
10. In the production environment, import all components and business events that were selectively exported in step 8.
11. In the production environment, import the new version of the example workflow that was selectively exported in step 7.
12. Go to Enterprise Management and detach the old version of the example workflow from the associated business process. Refer to documentation earlier in this chapter for the location of the example workflow.
13. Attach the new version of the example workflow to the associated business process.
14. If a new business event was imported in step 10, attach the business event to the associated business process in Enterprise Management. The business event must be attached so the new version of the example workflow can be triggered when necessary.
15. Save the changes.

Section II

Luminis Integration





Banner Workflow 8.0 provides the ability to integrate with Luminis Platform III.3.3.x or IV. This enhancement delivers integration with the SunGard Higher Education Portal, Collaboration, and Community Solutions to bring people together in a unified digital campus. Institutions can provide simple, personalized, and secure access to information so that administrators, faculty, and staff can perform everyday tasks with ease. This multifaceted enhancement is broken out into several key deliverables:

- Banner Workflow Product Tab Access
- Banner Workflow Channels
- Single Sign On via LDAP Authentication.

Banner Workflow Product Tab Access

Through the Banner Workflow Product Tab, Banner Workflow Users are provided with direct access (via the portal) to the comprehensive Banner Workflow product. As a feature of this integration the Banner Workflow product will adopt the Luminis look and feel when accessed through the portal environment. This allows for seamless integration between both products. For example:

Home

- Worklist
- Workflow Status Search
- Workflow Alerts

User Profile

- My Processes
- User Information
- Change Password

Administration

- Business Component Catalog
- Enterprise Management
- User Management
- Role Management
- Business Events
- Work Calendars
- In-process Monitoring

Worklist

Organization	Workflow	Activity	Priority	Created
Root.Camp1	New Gift 2006 Townsend, Joe Ready	EnterData	Normal	01-Nov-2006 02:43:04 PM
Root.Camp1	Address Change - Thomas, Al Ready	EnterData	Normal	01-Nov-2006 02:42:25 PM
Root.Camp1	Student Withdrawal - 200610 Jones, Pauline Ready	EnterData	Normal	01-Nov-2006 02:41:49 PM
Root	Approval Ready	Approve	Normal	01-Nov-2006 12:57:18 PM

1 - 4 of 4 First Previous Next Last Go to page: 1

Show Reserved Items

Add a Banner Workflow Tab

To Set up a Banner Workflow Tab:

1. Logon to Luminis.
2. Select **Content/Layout**.
3. Click **Add New Tab**.
4. Specify a name for the tab. This value will be the displayed value on the tab. For example:

Banner Workflow

5. Select the **Framed** tab type and supply the CPIP url. The URL field provides the CPIP URL back to Banner Workflow. For example:

`http://luminis.school.edu/cp/ip/login?sys=sctwf&api=workflow`

Where `luminis.school.edu` is the root of your Luminis server.

6. Select the desired position of the tab.

Click **Submit**.

Banner Workflow Channels

Part of Luminis Integration includes the delivery of three Banner Workflow channels. The channels provide portal access to key areas of Banner Workflow including:

- **Worklist** - The Worklist Channel displays as Banner Workflow User's worklist within the portal format. From the channel, workflow users may interact with, launch, and complete work items. Consistent with the traditional worklist access, as items are completed and when new tasks arise, work items are published and removed from the worklist accordingly.
- **Workflow Processes** - The Workflow Processes Channel provides Banner Workflow Users with the ability to directly/manually start workflow processes. This is the same functionality that is available via My Processes within Banner Workflow.
- **Workflow Alerts** - Geared towards workflow process owners and administrators, the Workflow Alerts Channel provides portal access to manage workflow instances. If a workflow process should encounter an error state, an alerts message is published to the process owners. The alert provides the owner/administrator with the ability to correct/ resolve the process error.

For example:

Workflow - Worklist

Organization	Workflow Name	Activity	Created	Details
Root.Camp1	New Gift 2006 Townsend, Joe	EnterData	01-Nov-2006 02:43:04 PM	
Root.Camp1	Address Change - Thomas, Al	EnterData	01-Nov-2006 02:42:25 PM	
Root.Camp1	Student Withdrawal - 200610 Jones, Pauline	EnterData	01-Nov-2006 02:41:49 PM	
Root	Approval	Approve	01-Nov-2006 12:57:18 PM	

[Refresh](#) [Show Entire Worklist](#) [Open Workflow](#)

Workflow - My Processes

Root
Grade Change
New Gifts

[Show My Processes Page](#) [Open Workflow](#)

Workflow - Alerts

Instance Name	Date	Process Name
Donor: Matthew Allen made a new gift # 0000785	01-Nov-2006 12:23:43 PM	CD Business Process - new gift - new name

The automated activity encountered the following error: ORA-00904: : invalid identifier

[Show Alerts Page](#) [Open Workflow](#)

